

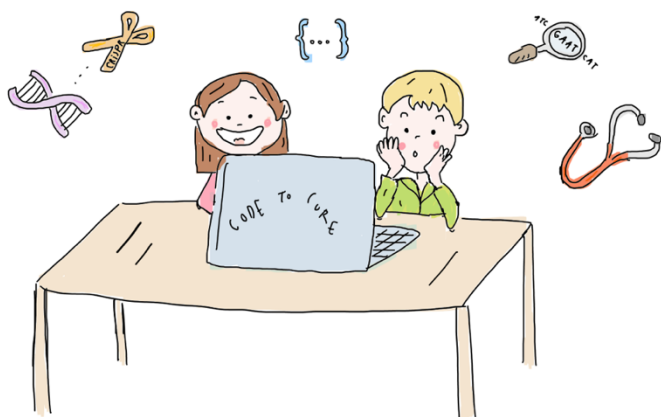


Norges miljø- og
biovitenskapelige
universitet

Masteroppgave 2023 30 stp

Fakultet for realfag og teknologi (REALTEK)

«Code to Cure»: Kan et kontekstbasert undervisningsopplegg påvirke elevenes holdninger til å programmere i biologi 2?



Illustrasjon: Mathea Veidal Kvernhaugen

Mathea Veidal Kvernhaugen

Lektorutdanning i realfag

«Code to Cure»: Kan et kontekstbasert undervisningsopplegg påvirke elevenes holdninger til å programmere i biologi 2?

English title: «Code to Cure»: Can a context-based teaching approach change students' attitudes towards programming in biology 2?

Forfatter:

Mathea Veidal Kvernhaugen

Veiledere:

Morten Munthe (hovedveileder) og Tonje Tomine Seland Strat (biveileder)

Masteroppgave

Lektorutdanning i realfag

30 studiepoeng

Fakultet for realfag og teknologi (REALTEK)

Institutt for lærerutdanning og utdanningsvitenskap

Norges miljø- og bioteknologiske universitet

Mai 2023

Forord

De fem siste årene har jeg vært så heldig å få kalle meg lektorstudent ved NMBU i Ås. Her har jeg lært mye om biologi, matematikk, didaktikk og pedagogikk. Arbeidet med nettopp denne oppgaven har vært en aldri så liten berg-og-dal bane. Utallige ganger har den holdt meg våken om nettene og motløs på diverse lesesaler og kafeer landet (og verden) over. Fremst av alt har den kostet mer tid og arbeid enn jeg noensinne kunne forestille meg. Likevel ville jeg aldri vært foruten. I arbeidet med oppgaven har jeg ikke bare lært mer om programmering i biologi, utviklingen av undervisningsopplegg og egen rolle som lærer. Jeg har lært mye om meg selv, og hvordan jeg kan få til de utroligste ting – bare jeg vil. Oppgaven har gitt meg mulighet til å være kreativ i utformingen av et undervisningsopplegg jeg ikke tør tenke på hvor mange timer jeg har brukt på å lage. Det har åpnet opp for en ny interesse og lidenskap i å forme undervisningsopplegg som bidrar til engasjement og mestring hos elever. Arbeidet med selve oppgaven har gitt meg refleksjoner jeg tror vil være svært verdifulle når jeg senere skal omsette teori til praksis i fremtiden som lærer.

Denne berg-og-dal-banen ville hatt betraktelige flere nedturer enn oppturer, hadde det ikke vært for alle de menneskene jeg har vært så heldig å ha rundt meg det siste året. Jeg vil rette en stor takk til familien min og venner som alltid er der for meg. Kjæresten min for at du rydda plass så jeg kunne få egen masterplass på kontoret ditt. Elevene som deltok i prosjektet og delte åpent om sine tanker og følelser til en lærer de ikke kjente. Faglærer som alltid har vært åpen og positiv til alle mine ideer, og bidratt med verdifulle innspill jeg nok ikke hadde klart meg uten.

Fremst av alt vil jeg rette en stor takk til veilederne mine – hovedveileder Morten Munthe og biveileder Tonje Tomine Seland Strat. Takk til Morten, for all den hjelpen jeg har fått helt siden jeg kom snublende inn på kontoret ditt og spurte om du kunne være veilederen min for snart et år siden. Hos deg har jeg fått hjelp, råd, oppmuntrende ord og ikke minst mulighet til å diskutere programmering med en av de få som er like engasjert i temaet som meg selv. Takk til Tonje, for all den jobben du har lagt i å støtte meg underveis i denne overveldende prosessen. Mer enn én gang har jeg kommet ut som et nytt og håpefullt menneske etter en veiledningstime med deg.

Tusen takk!

Mathea Veidal Kvernhaugen, mai 2023.

Sammendrag

Programmeringens «inntog» i den norske skolen som en del av Fagfornyelsen 2020, begrunnes gjerne med argumenter om et behov for at skolen skal ruste elever med kompetanser og ferdigheter som er viktige i dagens og fremtidens samfunn. Også i faget biologi legger læreplanene opp til at elevene skal programmere som en del av undervisningen (Utdanningsdirektoratet, 2020). Programmeringens inntog introduserer flere spørsmål om hvordan undervisere kan legge til rette for gode undervisningssituasjoner der programmering og fag møtes. Tidligere studier har vist at programmering i fagene kan by på både muligheter og utfordringer. En av utfordringene som presenteres, handler om elevenes holdninger og hvordan disse påvirker elevenes atferd, prestasjoner og opplevelser med programmering i undervisningen. Per i dag finnes det ennå lite forskning på programmering i biologi. Enda mindre finnes om elevenes *holdninger* til å programmere i faget, og hvordan disse kan stimuleres ved hjelp av et spesifikt undervisningsopplegg.

Min studie plasserer seg innenfor en undervisningskontekst med programmering i biologi 2, der jeg har undersøkt hva som kjennetegner elevens holdninger til å programmere i faget, og hvordan disse holdningene påvirkes av et kontekstbasert undervisningsopplegg utviklet for oppgaven. Studien er gjennomført i én biologi 2 klasse, og det er brukt en kombinasjon av både kvalitative og kvantitative datainnsamlingsmetoder med spørreskjema og fokusgruppeintervju. Datamaterialet har blitt analysert for å finne kjennetegn ved elevenes holdninger med vekt på engasjement, mestring og nytteverdi. Det er også gjort analyser for å undersøke elevenes opplevelser med undervisningsopplegget «Code to Cure», og hvordan opplegget har påvirket deres holdninger til å programmere i faget.

Studiens funn viser at elevenes holdninger i stor grad kjennetegnes av lavt engasjement, lav motivasjon og lav mestring i møte med programmeringsaktiviteter – generelt og i faget biologi 2. Videre har elevene flere forestillinger som stiller spørsmålstegn ved programmeringens nytteverdi i faget. Resultatene viser at tidligere erfaringer med programmering er svært avgjørende når elevenes holdninger dannes, og at det er flere faktorer som spiller inn på elevenes opplevelser. Videre har jeg funnet indikasjoner på at elevenes holdninger til å programmere i faget ble positivt påvirket av «Code to Cure», i form av en mer positiv følelsesmessig innstilling, økt mestring og en økt bevisstgjøring omkring programmeringens nytteverdi i og for faget. Resultatene kommer med flere implikasjoner for skolen og lærere som ønsker å stimulere elevenes holdninger og skape positive opplevelser med programmering i fag som biologi 2. Et annet bidrag er selve undervisningsopplegget, som representerer en enkeltstående programmeringsaktivitet som etter hvert vil kunne tas i bruk i også andre, norske biologiklasserom.

Abstract

The introduction of programming in Norwegian schools as part for Fagfornyelsen 2020, is often justified by the need to equip students with competencies and skills that are crucial in today's and future society. Biology is one of the subjects where students are required to program as part of their education (Utdanningsdirektoratet, 2020). This introduction raises several questions regarding how teachers can form and facilitate good teaching situations where programming and subjects meet. Previous studies have shown that the implementation of programming in already existing school subjects can present opportunities and challenges. One of the challenges presented in the literature, involves students' attitudes and how these affect student behavior, achievement, and overall experiences with programming as part of their education. Currently, there is still very little research regarding programming in biology. Even less is done on students' attitudes towards programming in subjects and how these can be stimulated by specific teaching approaches.

This particular study is situated within an educational context that involves programming in Biology 2. Here I have studied what characterizes students' attitudes towards programming, and how these attitudes are influenced by a specific, context-based teaching activity – developed for the project. The study was conducted in a single Biology 2 class, using a combination of qualitative and quantitative data collection methods, including surveys and focus group interviews. The data was analyzed to identify the characteristics of students' attitudes – focusing on engagement, mastery and usefulness. Analyses were also conducted to investigate students' experiences with the “Code to Cure” teaching activity and how it influences their attitudes towards programming in the subject.

The findings indicate that students' attitudes are mainly characterized by low engagement, low motivation, and low mastery when engaging in programming activities. Furthermore, the students have several beliefs that question the usefulness of programming in Biology 2. The results show that previous experiences with programming are crucial in forming student' attitudes, and that there are several factors influencing students' overall experiences. Additionally, indications were found that students' attitudes were positively influenced by “Code to Cure”, in the form of a more positive emotional disposition towards programming, increased mastery, and an awareness of the usefulness of programming in and for the subjects. The results have several implications for schools and teachers who wish to stimulate students' attitudes and create positive learning experiences with programming in subjects such as Biology 2. Another contribution is “Code to Cure” itself, a context-based teaching activity that eventually could be used in other Norwegian biology classrooms.

Innholdsfortegnelse

<i>Forord</i>	II
<i>Sammendrag</i>	IV
<i>Abstract</i>	V
<i>Innholdsfortegnelse</i>	VII
<i>Oversikt over figurer</i>	IX
<i>Oversikt over tabeller</i>	IX
<i>Forkortelser</i>	IX
Kapittel 1: Introduksjon	1
1. 1 <i>Bakgrunn for valg av tema</i>	1
1. 2 <i>Hensikt og forskningsspørsmål</i>	2
1.3 <i>Oppgavens struktur</i>	3
Kapittel 2: Teori	4
2. 1 <i>Holdninger</i>	4
2. 1. 1 <i>Holdningsmodeller</i>	4
2. 1. 2 <i>Holdningsbegrepet i denne oppgaven</i>	8
2. 2 <i>Programmering</i>	9
2. 2. 1 <i>Begrepet programmering</i>	9
2. 2. 2 <i>Programmering i biologi</i>	10
2. 2. 3 <i>Programmering i biologiundervisningen – læreplan og fagfornyelsen</i>	10
Kapittel 3: Litteratur	13
3. 1 <i>Programmering i undervisningen – utfordringer og muligheter</i>	13
3. 2 <i>Programmering i undervisningen – kjennetegn ved elevenes holdninger</i>	16
Kapittel 4: Metode	20
4. 1 <i>Undervisningsopplegget «Code to Cure»</i>	20
4. 1. 1 <i>Proessen med utviklingen av undervisningsopplegget</i>	20
4. 1. 2 <i>Beskrivelsen av gjennomføringen og undervisningsopplegget «Code to Cure»</i>	22
4. 2 <i>Metode og forskningsdesign</i>	24
4. 2. 1 <i>Studiens utvalg</i>	25
4. 2. 2 <i>Datainnsamlingsmetoder i studien</i>	26
4. 3 <i>Analyse</i>	32
4. 3. 1 <i>Kvantitativ analyse</i>	32
4. 3. 2 <i>Kvalitativ analyse</i>	34
4. 4 <i>Reliabilitet og validitet</i>	38
Kapittel 5: Resultater	42
5. 1 <i>RQ1 – Tidligere erfaringer og betydningen av disse på elevenes holdninger til å programmere</i>	42
5. 2 <i>RQ2 – Elevenes holdninger til å programmere i biologi</i>	45
Holdningsdimensjon A – Elevenes følelsesmessige innstilling til programmering	45
Holdningsdimensjon B – Elvenes oppfatning av egen kompetanse	46
Holdningsdimensjon C – Elevenes syn på programmering	47
5. 3 <i>RQ3 – elevenes opplevelser med undervisningsopplegget «Code to Cure»</i>	50

5.3.1 En sammenfatning av egne og faglærers observasjoner fra gjennomføringen	50
5.3.2 Elevenes generelle meninger om undervisningsopplegget «Code to Cure».....	51
5.3.3 Styrker og svakheter ved undervisningsopplegget	52
5.3.4 Undervisningsoppleggets effekt på elevenes holdninger til programmering i biologi 2.....	56
Kapittel 6: Diskusjon.....	61
6.1 Tidligere erfaringers betydning for elevenes holdninger til å programmere i biologi	62
6.2 Elevenes holdninger til å programmere i biologi 2	64
6.3 Erfaringer med undervisningsopplegget «Code to Cure».....	67
6.3.1 Elevenes opplevelser med undervisningsopplegget	67
6.3.3 Undervisningsoppleggets effekt på elevenes holdninger til å programmere i biologi 2.....	70
6.4 Studiens styrker og begrensninger.....	74
Kapittel 7: Avsluttende betraktninger	78
Referanser	81
Vedlegg.....	84
Vedlegg A: Godkjenning fra NSD.....	85
Vedlegg B: Code to Cure – læringssti	87
Vedlegg C: Code to Cure – link og utdrag fra undervisningsopplegget.....	94
Vedlegg D: Samtykkeskjema	100
Vedlegg E: Spørreskjema 1 (SS1).....	104
Vedlegg F: Spørreskjema 2 (SS2).....	111
Vedlegg G: Spørreskjema 3 (SS3)	117
Vedlegg H: Intervjuguide (med forslag til oppfølgingsspørsmål).....	119
Vedlegg I: Påstandene (gruppert under holdningsdimensjoner m/p-verdi).....	120
Vedlegg J: Utfyllende beskrivelse av tema (kvalitativ analyse)	121
Vedlegg K: Forslag til revidering av undervisningsopplegget – valg og begrunnelser.....	123

Oversikt over figurer

Figur 1: Den tredelte holdningsmodellen til Eagly og Chaiken (1993).	5
Figur 2: Det affektive perspektivet som beskrevet av McLeod (1992)	5
Figur 3: Holdninger som 4 evaluerende prosesser av Hannula (2002)	6
Figur 4: Holdningsdimensjonene av Di Martino og Zan (2010)	7
Figur 5: Utviklingen av undervisningsopplegget "Code to Cure"	20
Figur 6: Studiens datagrunnlag	27
Figur 7: Det abduktive rammeverket – tema og teori	36
Figur 8: Forskjell gjennomsnittsverdi (positive påstander) – pre og post-undersøkelsen	57
Figur 9: Forskjell gjennomsnittsverdi (negative påstander) – pre og post-undersøkelsen	58

Oversikt over tabeller

Tabell 1: Tidsramme og komponentene i undervisningsopplegget	23
Tabell 2: Oppgavens datakilder	25
Tabell 3: Antall og kjønnsfordeling i studiens utvalg	26
Tabell 4: Gruppering av påstander under holdningsdimensjonene	32
Tabell 5: Positive/negative påstander - definisjon	32
Tabell 6: Eksempel på kodingsprosessen	35
Tabell 7: Temaene under RQ1 med tilhørende definisjoner og eksempler utviklet i analyseprosessen	37
Tabell 8: Temaene under RQ2 med tilhørende definisjoner og eksempler utviklet i analyseprosessen	38
Tabell 9: Temaene under RQ3 med tilhørende definisjoner og eksempler utviklet i analyseprosessen	38
Tabell 10: Fag elevene har programmert i tidligere	42
Tabell 11: «Følelsesmessig innstilling» - gjennomsnitt påstander	45
Tabell 12: Koding av besvarelser som refererer til holdningsdimensjonen «Følelsesmessig innstilling»	46
Tabell 13: «Oppfatning av egen kompetanse» - gjennomsnitt påstander	47
Tabell 14: «Syn på programmering» - gjennomsnitt påstander	48
Tabell 16: Generelle meninger om undervisningsopplegget – gjennomsnitt påstander	51
Tabell 17: Styrker og svakheter ved undervisningsopplegget – gjennomsnitt påstander	52
Tabell 18: Styrker og svakheter ved undervisningsopplegget «Code to Cure» (kvalitative data)	55
Tabell 19: Effekt av undervisningsopplegget på holdninger – gjennomsnitt påstander	56
Tabell 20: Påstandene som viste signifikant forskjell (pre-test/post-test)	56
Tabell 21: Undervisningsoppleggets effekt på holdninger (kvalitative data)	58

Forkortelser

Forkortelse	Forklaring
LK20	Kunnskapsløftet 2020
NOU	Norges offentlige utredninger
Post-undersøkelsen	Datainnsamling som ble gjort <i>etter</i> gjennomføring av undervisningsopplegg
Pre-undersøkelsen	Datainnsamling som ble gjort <i>før</i> gjennomføring av undervisningsopplegg
RQ1	Forskningsspørsmål 1
RQ2	Forskningsspørsmål 2
RQ3	Forskningsspørsmål 3
SS1	Spørreskjema 1 (pre-undersøkelsen)
SS2	Spørreskjema 2 (post-undersøkelsen)
SS3	Spørreskjema 3 (post-undersøkelsen)

Kapittel 1: Introduksjon

1. 1 Bakgrunn for valg av tema

I dagens samfunn har teknologi en enorm innvirkning i våre liv – både på fritiden og i arbeid. Teknologien blir stadig en større og viktigere del av den verden vi lever, og jo mer denne teknologien utvikler seg og blir en mer integrert del av våre liv – jo viktigere blir det for oss mennesker å være utrustet med de ferdighetene og den kunnskapen som er nødvendige for å bruke denne teknologien på en god måte. En av skolens viktigste funksjoner, er å sørge for at elever får de ferdighetene og den kunnskapen som kreves i dagens og morgendagens samfunn (Kunnskapsdepartementet, 2017). En rapport levert av Ludvigsen-utvalget i 2015 understreker behovet for at skolen og skolefagene må endres for å imøtekomme fremtidens kompetansebehov (NOU 2015: 8, 2015). *Programmering* er en sentral del av denne teknologiske kompetansen, og argumenter for programmering i skolen knyttes derfor gjerne til et behov for ferdigheter innen programmering i dagens og morgendagens samfunn (Sevik, 2016).

Gjennom Fagfornyelsen 2020, har programmering hatt sitt inntog i flere av læreplanene for fag i skolen. Også i biologi, der elevene skal bruke programmering for å utforske ulike biologiske problemstillinger (Utdanningsdirektoratet, 2020). Pevzner og Shamir (2009) hevder at programmering og databehandling har en sentral rolle i dagens biologi, og viser til et behov for at utdanningen må holde følge med denne utviklingen i faget. Sevik (2016) hevder at en av mulighetene som oppstår ved å integrere programmering i de allerede eksisterende fagene i skolen, er at det gir elevene muligheter til å knytte teoretiske begreper opp mot en praktisk kontekst, og elevene kan få en forståelse av kompetansen i en større sammenheng. Potensialet ved å integrere programmering inn i allerede etablerte fag, støttes av Haraldsrud et al. (2020) som argumenterer for at programmering kan være et ledd i den dybdelæringen som står sentralt i Fagfornyelsen 2020.

I dag (våren 2023) finnes det lite forskning på bruk av programmering i undervisningen på videregående skole, og spesielt lite finnes om programmering inn i biologifaget. Tidligere forskning fra Norge har sett på programmering inn mot faget matematikk, og studiene beskriver hvordan implementeringen av programmering inn i faget kan by på flere utfordringer – både for lærere og elever (Andersen, 2022; Munthe, 2022; Zukanovic, 2021). Dette sammenfaller med funn fra andre studier som ser på programmering i andre land eller andre undervisningskontekster, som Vinnervik (2021), Eliassen et al. (2021) og Sengupta et al. (2013). En av utfordringene handler om elevenes holdninger og hvordan disse kan påvirke elevenes fremtidige møter med programmering i fagene (Eliassen et al., 2021; Forrester et al., 2022). Andersen (2022) fant i sin masteroppgave at elevenes holdninger til å programmere i

matematikk kjennetegnes av lav motivasjon og lav mestringsfølelse. Dette samsvarer godt med det inntrykket jeg selv sitter med etter samtaler med lærere og elever i praksis.

Til tross for at det i litteraturen finnes flere argumenter for programmeringens nytteverdi inn mot fagene, viser utfordringene som oppstår til et behov for mer forskning på implementeringen av programmering i undervisningen. Munthe (2022) argumenterer for at det er et behov for mer forskning på hvordan lærere kan lage undervisningsopplegg som minsker disse utfordringene, slik at kompetansen kan bidra til økt forståelse i faget. Mangelen på et større forskningsfelt om programmering i biologi, poengterer nødvendigheten for å gjennomføre studier som undersøker integreringen av programmering inn i faget. Samtidig finnes det generelt lite forskning på *holdninger* til programmering – og særlig fra elevenes perspektiv. Dette «gapet» i forskningslitteraturen understreker et behov for videre forskning på området, og utgjør en av de viktigste begrunnelsene for valg av tema til denne masteroppgaven.

1. 2 Hensikt og forskningsspørsmål

Gjennom studien ønsker jeg økt innsikt i hva som kjennetegner biologi 2 elevers holdninger til programmering i biologi, og hvilke faktorer som blir avgjørende når disse holdningene formes. Med denne informasjonen som «bakteppe», ønsker jeg å undersøke hvordan et bestemt undervisningsopplegg kan påvirke elevenes holdninger til verktøyet i faget. I den forbindelse utviklet jeg det kontekstbasert undervisningsopplegget «Code to Cure», der programmering integreres i et undervisningsopplegg innenfor temaet bioteknologi. Oppgavens hensikt blir å undersøke hvordan det kontekstbaserte undervisningsopplegget «Code to Cure» påvirker biologi 2 elevers holdninger til å programmere i faget.

Videre har jeg spesifisert tre forskningsspørsmål, der de to første legger vekt på kjennetegn ved biologi 2 elevers holdninger til å skulle programmere i faget og hva som former disse. Det tredje forskningsspørsmålet er knyttet opp mot oppgavens hensikt, og tar utgangspunkt i det spesifikke undervisningsopplegget og en eventuell effekt av dette på elevenes holdninger. Mine forskningsspørsmål blir som følger:

Forskningsspørsmål 1 (RQ1): *Hvilke tidligere erfaringer har elevene med programmering i skolen, og hvilken betydning har erfaringene for elevenes holdninger til å programmere i biologi?*

Forskningsspørsmål 2 (RQ2): *Hva kjennetegner elevenes holdninger til å programmere i biologi 2?*

Forskningsspørsmål 3 (RQ3): *Hvordan påvirkes elevenes holdninger til å programmere i biologi 2 av det kontekstbaserte undervisningsopplegget «Code to Cure»?*

Ved å belyse forskningsspørsmålene ovenfor, kan min masteroppgave forhåpentligvis bidra med perspektiver i diskusjonen om veien videre, og være et bidrag inn mot å forstå hvordan lærere kan støtte og engasjere elever i situasjoner der programmering og biologi møtes i skolen.

1.3 Oppgavens struktur

I kapittel 2 følger en gjennomgang av teori som er relevant for å belyse mine forskningsspørsmål. I litteraturgjennomgangen (kapittel 3) vil jeg forsøke å orientere meg i landskapet av tidligere litteratur og forskning, og forsøke å besvare spørsmålet «Hvor i dette landskapet vil jeg plassere nettopp mitt prosjekt?». Kapittel 4 utgjør oppgavens metodekapittel. Her vil jeg først gi en beskrivelse av undervisningsopplegget «Code to Cure» og utviklingen av opplegget. Videre presenterer jeg utvalget, og legger frem og begrunner de metodiske valgene som er gjort i datainnsamlingen og analyseprosessen.

Resultatkapitlet (kapittel 5) viser resultatene fra datainnsamling og analysearbeid. I kapittel 6 drøftes resultatene i lys av teori og tidligere forskning. Avslutningsvis vil jeg i kapittel 7 komme med noen avsluttende betraktninger vedrørende studiens funn og forslag til videre forskning.

Jeg vil presisere at deler av masteroppgaven er skrevet med utgangspunkt i tidligere arbeid som er levert i forbindelse med det masterforberedende faget PPUT301. Dette gjelder deler av teori- og litteraturgjennomgang. Der det er brukt deler av tekst direkte, vil dette underveis være markert med fotnoter i henhold til retningslinjer fra ILU ved NMBU.

Kapittel 2: Teori

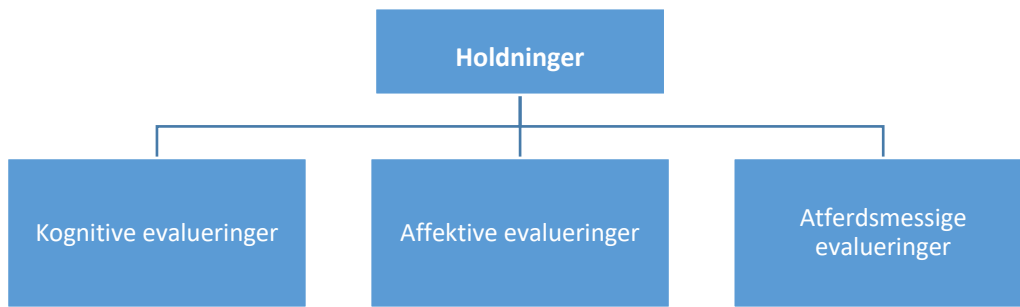
Teori som er av sentral betydning i mitt forskningsprosjekt vil her presenteres fordelt på to domener – et om holdninger og det affektive perspektivet, det andre om programmering og programmeringens rolle i faget biologi. Teorien og begrepene som presenteres vil videre bli brukt som fundament for å belyse og drøfte studiens tematikk og funn.

2. 1 Holdninger

Holdningsbegrepet er et komplekst begrep det er vanskelig å definere entydig, og forskere anvender ulike definisjoner og modeller for å forklare hva holdninger er og hva de faktisk innebærer. Det flere forskere imidlertid viser stor enighet om, er viktigheten av å forstå elevenes holdninger og hvordan disse formes. Lærere beskriver ofte lavt presterende og umotiverte elever ved å si at de har «dårlige holdninger» (Di Martino & Zan, 2010, s. 10), og i forskningslitteraturen viser flere til en sammenheng mellom elevers holdninger og prestasjoner i fag (Baser, 2013; Di Martino & Zan, 2010; Erol, 2020; McLeod, 1992). Dette har også vist seg gjeldende i undervisningssituasjoner med programmering (Baser, 2013; Erol, 2020). Andre fremhever hvordan affektive faktorer, deriblant holdninger, er viktige å forstå fordi de spiller en sentral rolle i forståelse og læring (McLeod, 1992). I dette kapitlet vil jeg redegjøre for hvordan holdningsbegrepet kan forstås i lys av ulike teoretiske perspektiver. Til slutt vil jeg presentere den definisjonen som danner utgangspunktet videre i oppgaven.

2. 1. 1 Holdningsmodeller

Eagly og Chaiken (1993) definerer holdninger som «a psychological tendency that is expressed by evaluating a particular entity with some degree of favor or disfavor» (s. 1). Forfatterne presenterer videre en tredelt definisjon av holdningsbegrepet, hvor holdninger formes på bakgrunn av kognitive, affektive og atferdsmessige evalueringer eller «prosesser». Den kognitive komponenten av holdninger innebærer en persons *tro, tanker og forestillinger* om et objekt, mens den affektive komponenten referer til *følelsene* som knyttes til objektet. Eksempelvis kan en gitt stimulus fremkalle en positiv eller negativ følelse hos en elev, og ved gjentatte lignende stimuleringer kan eleven etter hvert knytte disse følelsene til selve objektet - som igjen kan utvikle seg til holdninger. Den atferdsmessige komponenten involverer en persons *handlinger og atferd* i møte med objektet. Slik argumenterer Eagly og Chaiken (1993) for at holdninger blir dannet, bevisst eller ubevisst, på bakgrunn av minst én av disse evalueringene, slik det er illustrert i figur 1.



Figur 1: Den tredelte holdningsmodellen til Eagly og Chaiken (1993).

Der modellen til Eagly og Chaiken (1993) beskriver holdninger generelt, uavhengig av kontekst og fag, er det utviklet flere holdningsmodeller for å forstå holdninger i spesifikke fag. I forskningslitteraturen er det flere som trekker linjer mellom programmering og faget matematikk, slik som Munthe (2022), som hevder at matematikk og programmering har mye til felles. Forrester et al. (2022) hevder at følelser som angst og bekymring for programmering, kan relatere til matematikkangst, gjennom det forfatteren kaller «quantitative anxiety» (s. 2). På bakgrunn av at det finnes mer forskning på holdninger i faget matematikk, vil det videre i kapitlet presenteres noen relevante holdningsmodeller fra matematikkfaget.

Med utgangspunkt i matematikdidaktikken presenterer McLeod (1992) en modell for sammenhengen mellom det affektive perspektivet, holdninger, følelser og oppfatninger. Forfatteren beskriver det affektive perspektivet som «a wide range of beliefs, feelings and moods that are beyond the domain of cognition» (McLeod, 1992, s. 576), der aspektene følelser, holdninger og forestillinger er *undergrupper* av det affektive domenet. De tre undergruppene likestilles, og graderes etter hvor stabile og intense de er, illustrert i den forenklete versjonen av holdningsmodellen, figur 2.

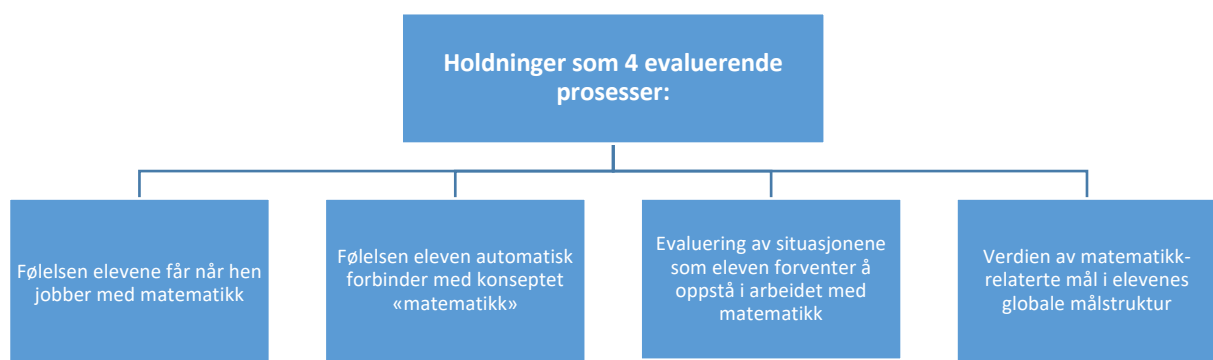


Figur 2: Det affektive perspektivet som beskrevet av McLeod (1992)

Av modellen ser vi at McLeod (1992) graderer oppfatninger som den mest stabile, minst intense og mest kognitive komponenten, og følelsene som de minst stabile, mest intense og minst kognitive. Holdninger

blir plassert midt mellom disse, og defineres som de negative og positive affektive responsene et individ har av *moderat intensitet og stabilitet* mot en bestemt kontekst. Videre hevder han at holdninger til matematikken kan skapes på to måter; 1) ved at allerede eksisterende holdninger til faget overfører til andre deler av faget, eller 2) gjennom gjentatte opplevelser til konkrete deler av matematikken. Slik kan gjentatte følelsesmessige reaksjoner utvikle seg til mer stabile følelser i form av *holdninger*. Eksempelvis kan en elev med gjentatte opplevelser av å være frustrert i matematikktimene etter hvert utvikle *mer stabile*, negative holdninger til faget. Dette minner om Eagly og Chaiken (1993) sin forståelse av hvordan holdninger kan skapes på bakgrunn av affektive evalueringer, slik det er beskrevet tidligere. Modellen til McLeod (1992) plasserer imidlertid holdningsbegrepet som *en del av* det affektive perspektivet, for å beskrive affektive responser et individ har av *moderat intensitet og stabilitet*. McLeod (1992) argumenterer for at holdninger ikke er faste, men gjenstand for endring basert på stadig ny informasjon og nye erfaringer. Videre argumenterer McLeod (1992) for at affektive faktorer er viktige å forstå fordi de spiller en sentral rolle i forståelse og læring.

Definisjonen til McLeod (1992), som beskriver holdninger som affektive responser av moderat intensitet og stabilitet, støttes av Hannula et al. (2004). Likevel kritiserer Hannula et al. (2004) modellen til McLeod (1992) for å være for enkel og for lite nyansert, ved at den blant annet ikke søker forklaring på *hvorfor* disse reaksjonene oppstår, og ved at den ikke tar høyde for andre relevante begreper som motivasjon, interesse og verdier når holdninger til elever skal identifiseres. På bakgrunn av dette, presenterer Hannula (2002) et rammeverk for å analysere holdninger basert på fire evaluerende prosesser om følelser, assosiasjoner, forventninger og verdier. De fire prosessene er gjengitt i figur 3, der de er oversatt av meg.

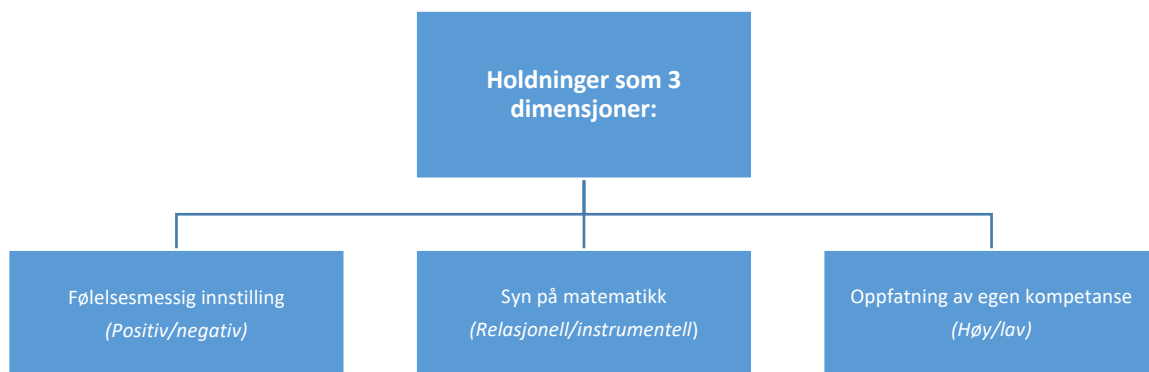


Figur 3: Holdninger som 4 evaluerende prosesser som beskrevet av Hannula (2002, s. 26). Oversatt fra engelsk av meg.

Holdningsmodellen TMA (three-dimensional Model for Attitude) er en mer sammensatt matematikdidaktisk holdningsmodell utviklet av Di Martino og Zan (2010) som beskriver holdninger gjennom de tre dimensjonene: 1) Følelsesmessig innstilling, 2) Oppfatning av egen kompetanse og 3)

Syn på matematikk (oversatt til norsk av meg). Dimensjonene har flere likhetstrekk med prosessene i rammeverket til Hannula (2002), og er illustrert i figur 4. Modellen ble empirisk utviklet i en studie om elevenes forhold til matematikk, hvor resultatene viste at nærmest alle elevene refererte til minst én av de tre dimensjonene i beskrivelsene av deres forhold til faget. Forfatterne hevdet at elevenes følelser til matematikk, syn på faget og oppfatning av egen matematisk kompetanse ble avgjørende for hvilke holdninger elevene hadde til matematikkfaget.

Til forskjell fra den matematikdidaktiske holdningsmodellen til McLeod (1992), der dimensjonene følelser, holdninger og forestillinger likestilles, presenterer Di Martino og Zan (2010) en modell med dimensjonene følelser og forestillinger som *en del av* holdningsbegrepet. I tillegg inkluderer modellen dimensjonen som omhandler elevenes syn på egen kompetanse, en dimensjon som i modellen til McLeod (1992) ble inkludert i komponenten *oppfatninger*. I artikkelen til Di Martino og Zan (2010), ble de tre dimensjonene redusert til dikotomier: følelsesmessig innstilling (positiv/negativ), syn på matematikk (relasjonell/instrumentell) og oppfatning av egen kompetanse (høy/lav). De ulike holdningsfaktorene henger nøye sammen og påvirker hverandre gjensidig, slik at dersom man tar sikte på å endre én av dimensjonene, kan dette påvirke de andre dimensjonene. Di Martino og Zan (2010) argumenterer for at en forståelse av dimensjonene kan hjelpe pedagoger med å forme undervisning som bedrer elever og studenters holdninger til matematikk, slik at de opplever engasjement og mestring i faget.



Figur 4: De tre holdningsdimensjonene beskrevet av Di Martino og Zan (2010), oversett fra engelsk til av meg.

Til tross for at det i forskningslitteraturen finnes forskjellige definisjoner og modeller for å forklare holdninger, er det enighet blant forskerne om at holdningsbegrepet er omfattende og komplekst å definere. Den tredelte modellen presentert av Eagly og Chaiken (1993) skiller mellom kognitive, affektive og atferdsmessige evalueringer som komponenter av holdninger. Modellen til McLeod (1992)

representerer en annen tilnærming, der holdninger forstås som en del av det affektive domenet. Både Eagly og Chaiken (1993) og McLeod (1992) understreker hvordan holdninger kan oppstå på bakgrunn av tidligere affektive evalueringer eller responser mot et objekt. Rammeverket til Hannula et al. (2004) og Di Martino og Zan (2010) åpner opp for mer nyanserte beskrivelser av elevers holdninger til matematikk, og beskriver disse i form av evaluerende prosesser (Hannula et al., 2004) eller holdningsdimensjoner (Di Martino & Zan, 2010).

2. 1. 2 Holdningsbegrepet i denne oppgaven

Som definisjon til grunn for begrepene «det affektive perspektivet» og «holdninger» velger jeg å ta utgangspunkt i en kombinasjon av to av de presenterte holdningsmodellene. Definisjonen støtter seg på McLeod (1992) sin forståelse av holdninger som en del av det affektive perspektivet, og som begrep for å beskrive de responsene et individ har av *moderat intensitet og stabilitet*. Definisjonen kombineres med de tre holdningsdimensjonene til Di Martino og Zan (2010), for å gi en mer nyansert beskrivelse av hvilke underkomponenter som er avgjørende for elevers holdninger. Videre tilpasses definisjonen slik at den omhandler *holdninger til programmering*, i stedet for til faget matematikk. Følgende definisjon representerer standpunktet som er tatt i forbindelse med denne oppgaven:

Elevenes holdninger til å programmere, omfatter de responsene elevene har av moderat intensitet og stabilitet ovenfor det å programmere generelt og i faget biologi 2. Holdningene er en del av det affektive perspektivet, og kan beskrives gjennom de tre holdningsdimensjonene: følelsesmessig innstilling, oppfatning av egen kompetanse og syn på programmering. De tre dimensjonene henger nøye sammen, og påvirker hverandre gjensidig.

For å beskrive de tre holdningsdimensjonene vil jeg ta utgangspunkt i definisjonene fra Di Martino og Zan (2010) og tilpasse de slik at de beskriver holdninger til programmering. *Følelsesmessig innstilling* refererer til et individs følelsesmessige/affektive respons til programmering. Dimensjonen inkluderer følelser som engasjement, motivasjon, frustrasjon og uro/bekymring, og påvirkes i stor grad av tidligere erfaringer med verktøyet. Dimensjonen henger tett sammen med de to andre dimensjonene, da en negativ følelsesmessig innstilling ofte oppstår på bakgrunn av oppfatning av egen kompetanse som lav eller negativt syn på programmering (Di Martino & Zan, 2010). *Oppfatning av egen kompetanse* refererer til et individs tro på egne evner til å mestre oppgaver med programmering. Dimensjonen inkluderer tro på egne ferdigheter, oppfatning av egen kunnskap og evne til å lære programmering. Oppfatning av egen kompetanse som *høy*, kan bidra til økt selvtillit og utholdenhet i problemløsning ved programmeringsaktiviteter (Di Martino & Zan, 2010). *Syn på programmering* refererer til et individs

forestillinger og/eller oppfatninger av programmering og programmeringens rolle i samfunnet. Dimensjonen inkluderer programmeringens nytteverdi og forholdet mellom programmering og andre fagdisipliner (Di Martino & Zan, 2010).

Modellene til McLeod (1992) og Di Martino og Zan (2010) er videre brukt som utgangspunkt i utviklingen av det abduktive rammeverket presentert i oppgavens analysedel.

2. 2 Programmering

I dette kapitlet vil jeg redegjøre for begrepet *programmering*, som utgjør det andre sentrale domenet i denne studien. Jeg vil kort redegjøre for programmeringens stadig viktigere rolle inn mot feltet biologi, før jeg avslutningsvis presenterer situasjonen i dagens biologiundervisning etter Fagfornyelsen 2020.

2. 2. 1 Begrepet programmering

Programmering som begrep kan forstås på flere ulike måter. For å kunne presentere definisjonen som danner utgangspunktet videre i denne oppgaven, vil jeg kort gjøre rede for noen av de definisjonene som finnes. I 2016 publiserte Sevik (2016) fagnotatet «Programmering i skolen» for Udir og Senter for IKT i utdanningen. Her definerer Sevik (2016) programmering som en *prosess*, mer enn kun det å skrive programkode som instruksjoner til en datamaskin. I denne definisjonen blir programmering prosessen med å skrive kode, beskrive hva programmet skal gjøre og designe løsninger. Videre understreker Sevik (2016) at prosessen med å komme frem til koden er like viktig som selve kodingen.

Statped (2021) skiller mellom begrepene programmering og koding. Programmering er beskrevet som en prosess der komplekse problemer brytes til mindre problemer og løses på en systematisk måte. Koding, derimot, refererer til de instruksene som gis til datamaskinen, men omfatter ikke prosessene med å komme frem til og utforske disse instruksene videre. Felles med Sevik (2016), bruker Statped (2021) programmering som begrep om situasjonene der det handler om å utforske, skape og bruke programmering som problemløsningsmetode. Elementene er sentrale i problemløsningsmetoden *algoritmisk tenkning* som vektlegges i flere av læreplanene (Sevik, 2016).

Sanne et al. (2016) gir en annen definisjon av begrepet programmering: "Programmering vil si å bryte et gitt problem ned i et sett av kommandoer og så får en datamaskin til å utføre disse kommandoene" (s. 18). Definisjonen vektlegger datamaskinens rolle, og utelater på denne måten det vi kaller analog programmering - programmering uten datamaskin. Videre i oppgaven har jeg vil jeg ta utgangspunkt i

definisjonen til Sevik (2016), da den ved å vektlegge aspekter ved utforskning, problemløsning og feilsøking, kan sies å gå en mer «helhetlig» forståelse av begrepet programmering.

2. 2. 2 Programmering i biologi

Inkluderingen av programmering i læreplanene for biologi, kan forklares som en reaksjon på programmeringens økende betydning innenfor det biologiske feltet. Forskere som Pevzner og Shamir (2009) beskriver hvordan databehandling har hatt en dyptgripende innvirkning på feltet biologi, og at biologiundervisningen må holde tritt med de pågående endringene som skjer i faget. Flere fremhever den stadig økende nytteverdien av programmering i faget, slik som Rahn et al. (2019) og David (2021):

«The rise of «big data» within the biological sciences has resulted in an urgent demand for coding skills in the next generation of scientists». (David, 2021)

“The biological sciences are becoming increasingly reliant on computer science and associated technologies to quickly and efficiently analyze and interpret complex data sets. Introducing students to data analysis techniques is a critical part of their development as well-rounded, scientifically, literature citizens” (Rahn et al. (2019), s. 649)

Forskerne Rahn et al. (2019) og David (2021) fremhever programmeringens betydning i faget ved å vise til programmeringens nytteverdi i behandling og analyse av komplekse og store datasett. På bakgrunn av behovet for kompetanser innen programmering i faget, fremhever de viktigheten av en utdanning som sikrer at fremtidige biologer blir utstyrt relevante kompetanser og ferdigheter.

2. 2. 3 Programmering i biologiundervisningen – læreplan og fagfornyelsen

På oppdrag fra Udir, leverte Ludvigsen-utvalget i 2015 NOU-en *Fremtidens skole - Fornyelse av fag og kompetanser*. Utvalget understreket behovet for at skolen og skolefagene må endres for å imøtekomme fremtidens kompetansebehov (NOU 2015: 8, 2015). Den teknologiske utviklingen i samfunnet går stadig raskere, og denne endringen må gjenspeiles i den undervisningen elevene får i skolen (Sanne et al., 2016). Programmering er en sentral del av denne teknologiske kompetansen, og ferdigheter innen programmering anses som viktige i dagens og morgendagens samfunn. Senter for IKT i utdanningen uttalte i 2016 at "Argumenter for programmering i skolen knyttes gjerne til nødvendige ferdigheter for

det 21. århundre, fremtidige behov for kompetanse i næringslivet og evne til å forstå hvordan et stadig mer digitalisert samfunn fungerer" (Sevik, 2016).¹

I dag er programmering gjennom Fagfornyelsen 2020 blitt en integrert del av de allerede eksisterende fagene i skolen – også i biologi². Dette til tross for at en ekspertgruppe oppnevnt av Utdanningsdirektoratet i 2016 i sin rapport om teknologi og programmering i grunnskolen, kom med anbefalingen om at det heller burde opprettes et nytt obligatorisk teknologifag i grunnskolen. Dette faget burde være et praktisk fag der elevene fikk tilegne seg kompetanser innenfor både teknologi og programmering (Sanne et al., 2016). Videre argumenterer arbeidsgruppa for at det å integrere programmering i allerede etablerte skolefag ikke er godt nok, og at elevene trenger kompetanse i programmering på programmeringens *egne premisser*, og ikke kun for å lære andre fag (Sanne et al., 2016)³.

Til tross for at det er to ulike tilnærminger til innføringen av kompetansen i skolen, peker Sevik (2016) på en mulighet som oppstår når man knytter programmeringen opp mot spesifikke fag i skolen. Dersom elevene kan kombinere ferdigheter i fag med ferdigheter i programmering, vil dette kunne gi elevene mulighet til å knytte teoretiske begreper opp mot en praktisk kontekst. Slik kan elevene få forståelse av kompetansen i en større sammenheng. Til tross for at forfatterne bruker matematikkfaget som et eksempel her, er det nærliggende å tro at lignende fordeler kan gjelde for andre fag i skolen. Denne muligheten støttes av Haraldsrud et al. (2020), som argumenterer for at programmering kan være et ledd i den dybdelæringen som står sentralt i Fagfornyelsen 2020⁴.

I biologi står programmering nevnt i den overordnede delen av læreplanen (Utdanningsdirektoratet, 2020). Her skal elevene bruke programmering som verktøy i utforskningen av biologiske problemstillinger:

*«Digitale ferdigheter i biologi inneber å bruke digitale ressursar til å utforske, illustrere, **programmere**, modellere og simulere biologiske system og prosessar»*

(Utdanningsdirektoratet, 2020)

Læreplanmålet spesifiserer hverken *når* eller på *hvilken måte* programmering skal implementeres i faget, og dette blir i stor grad derfor opp til den enkelte lærere å vurdere. Det er likevel flere sentrale

¹ Deler av avsnitt er hentet fra tidligere selvskrevet oppgave i PPUT301

³ Deler av avsnitt er hentet fra tidligere selvskrevet oppgave i PPUT301.

⁴ Deler av avsnitt er hentet fra tidligere selvskrevet oppgave i PPUT301.

tema i undervisningsfaget biologi, som kan sies å åpne opp for bruk av programmering - som for eksempel behandlingen av «big data»⁵, temaer innen genetikk og sekvensering av proteiner (Munthe, 2022). Eliassen et al. (2021) skriver i sin forskningsartikkel hvordan innføringen av programmering i skolen som følge av de nye læreplanene, begrunnes med at utdanningen må møte kompetansebehovet for fremtiden (NOU:2014:7, 2014), slik at dagens og fremtidens biologer utvikler relevante og nødvendige kompetanser (David, 2021; Pevzner & Shamir, 2009; Rahn et al., 2019).

⁵ Big data i biologi refererer her til store og komplekse datasett som inneholder informasjon om biologiske systemer og fenomener – som for eksempel DNA-sekvenser. Stephens, Z. D., Lee, S. Y., Faghri, F., Campbell, R. H., Zhai, C., Efron, M. J., Iyer, R., Schatz, M. C., Sinha, S. & Robinson, G. E. (2015). Big data: astronomical or genomics? *PLoS biology*, 13(7), e1002195.

Kapittel 3: Litteratur

I dette kapitlet følger en sammenfatning av tidligere litteratur og forskning som blir relevant for studiens analyse og diskusjon. Først vil jeg presentere noen av de forsøkene som er gjort på å innføre programmering i undervisningen, og hvilke muligheter og utfordringer en slik integrering kan by på. Videre følger en presentasjon av et utvalg artikler som mer spesifikt sier noe om hva som kjennetegner elevers og studenters *holdninger* til det å programmere som en del av undervisningen.

I den påfølgende gjennomgangen vil jeg i tillegg trekke frem erfaringer som er gjort, både med tanke på å styrke muligheter og minske utfordringer når programmering implementeres i undervisningen, men også erfaringer som er vist seg positive når elevenes holdninger til programmering ønskes endret. Flere av erfaringene blir relevante i oppgavens metodekapittel, der prosessen med utformingen av undervisningsopplegget presenteres.

3. 1 Programmering i undervisningen – utfordringer og muligheter

Med et ønske om å kunne utvikle retningslinjer til lærere som skal inkludere programmering inn i undervisningen, ser en studie av Sengupta et al. (2013) nærmere på programmering i realfagene naturfag og matematikk. Artikkelen bygger på en mindre studie der forskerne har gjennomført et bestemt undervisningsopplegg i en 6. klasse i USA (n= 24). Innledningsvis presenterer Sengupta et al. (2013) noen av mulighetene som kan oppstå ved å implementere programmering i undervisningen av andre fag, som at programmering kan bidra til økt læringsutbytte og være et verdifullt verktøy for å forstå komplekse konsepter i naturfag og matematikk. Sengupta et al. (2013) påpeker likevel at det eksisterer flere utfordringer, som at programmering i seg selv er vanskelig å lære og at elevene kan oppleve det som utfordrende å skulle lære både programmering og fag samtidig. Dette kan resultere i at elevene i programmeringsaktiviteter oftere møter utfordringer relatert til selve programmeringen, enn til faget i seg selv. Forfatterne understreker behovet for at undervisere må tenke nytt i utformingen av programmerings-baserte læringssituasjoner i realfagene, og presenterer i sin diskusjon flere retningslinjer som kan bidra positivt i en slik endring; Opplegget burde være enkelt å bruke for både elever og lærere med lite programmeringserfaring, godt integrert i eksisterende læreplaner og utformet slik at det tilrettelegger for aktiviteter med lav inngangsterskel og stor takhøyde. For å stimulere til økt læringsutbytte og engasjement, hevder forfatterne at det er en fordel dersom opplegget er kontekstbasert og virkelighetsnært. Sengupta et al. (2013) understreker viktigheten av at *faget* kommer i fokus, og at programmering derfor må integreres der det er hensiktsmessig i undervisningen.

I perioden 2017 - 2020 gjennomførte Munthe (2022), som en del av sin doktorgradsavhandling, flere undervisningsopplegg med programmering i matematikk R1 og R2 på en videregående skole i Norge. I sin avhandling presenterer forfatteren flere utfordringer som kan oppstå når programmering implementeres i skolefagene. Blant utfordringene er elevenes manglende forkunnskaper, lærernes mangel på programmeringsteknisk og programmeringsdidaktiske kompetanse og tvetydighet i læreplanene. Munthe (2022) fremhever utfordringene lærere står ovenfor når de skal utvikle undervisningsopplegg som kombinerer fag og programmering på en god måte. Som i artikkelen til Sengupta et al. (2013) fremheves utfordringene ved at elevene må lære seg programmering og fag samtidig, og Munthe (2022) beskriver hvordan dette kan føre til at elever blir frustrerte og utvikler «poor visions» for hva programmering er (s. 23).

På samme måte som Sengupta et al. (2013), presenterer Munthe (2022) i sin diskusjon et sett med anbefalinger for lærere som utformer undervisning med programmering i faget. Forfatteren hevder at flere av disse kan være gjeldende i også andre fag enn matematikk. Som Sengupta et al. (2013), finner han at det er en fordel om programmeringen har en tydelig kobling opp mot faget slik at faget kommer i fokus og at oppgavene burde ha en lav inngangsterskel og høy takhøyde slik at flest mulig opplever mestring. Opplegget burde videre bidra til utforskende samtaler mellom elevene. Det kan være en fordel dersom programmeringen inkluderes etter at elevene allerede har lært om et tema, og at undervisningsopplegget legger til rette for at de hindringene som oppstår i størst mulig grad er koblet opp mot matematikkfaget, og minst mulig opp mot programmeringen. Dette kan være viktig for å unngå at kombinasjonen med programmering og matematikk samlet blir for komplisert for elevene. Avslutningsvis konkluderer Munthe (2022) med at programmeringen kan bidra positivt inn i fagene, og at læringsverktøyet har et stort potensial i undervisning. Viktig blir det likevel å anerkjenne de utfordringene som både lærere og elever støter på i møte med programmering, og ta dette i betraktning når undervisningen planlegges og gjennomføres.

Som del av sitt masterprosjekt undersøkte Waters (2020) hvorvidt programmering kunne bidra til økt dybdelæring i fysikk 1. I den forbindelse utviklet og gjennomførte han et undervisningsopplegg i to fysikk 1 klasser på en videregående skole. Studiens funn viste at elevene i større grad så sammenhenger og fikk eierskap til de fysikkfaglige problemstillingene når de programmerte som del av aktiviteten. På samme måte som Sengupta et al. (2013), fant Waters (2020) funn som indikerer viktigheten å gi elevene oppgaver som oppleves nyttige og virkelighetsnære. Forfatteren fant at disse aspektene ved undervisningsopplegget var avgjørende for elevenes eierskapsfølelse. Videre fant Waters (2020) at elevene som brukte mye kapasitet på å håndtere utfordringer knyttet til programmering, fikk mindre overskudd til å reflektere over fysikkfaglige begreper og konsepter. Denne utfordringen kan virke

hemmende for dybdelæringen i faget, og er et viktig aspekt ved integreringen av programmering inn i andre fag, som forfatteren finner støtte for i litteraturen (Munthe, 2022; Sengupta et al., 2013)⁶.

Flere av mulighetene og utfordringene med å implementere programmering i undervisningen presenteres i doktorgradsavhandlingen til Vinnervik (2021). Den overordnede hensikten i studien er å undersøke hvilke omstendigheter som er av betydning for lærernes arbeid med å innføre programmeringen i undervisningen, i dette tilfellet i Sverige. Vinnervik (2021) argumenterer for at det er viktig å undersøke hvilke utfordringer som oppstår, slik at forskere og lærere kan utvikle strategier for å overvinne disse og skape god undervisning med programmering. Av utfordringer trekker han frem lærernes mangel på både programmeringsteknisk og programmeringsdidaktisk kompetanse. Forfatteren argumenterer for at programmering er både vanskelig og tidkrevende å lære seg. En annen utfordring er *tid*, både ved at programmering kan ta tid fra den «vanlige» undervisningen i faget, men også i form av at lærerne har lite tid til å lære seg programmering og å utvikle gode undervisningsopplegg. Til tross for flere utfordringer, hevder Vinnervik (2021) likevel at programmering åpner opp for flere muligheter i fagene; både som verktøy for å forstå faget, men også for å gjøre undervisning interessant og relevant. Han viser til et behov for mer forskning på hvordan undervise programmering, og hvordan implementere programmering inn i de allerede eksisterende fagene på en god måte. Vinnervik (2021) argumenterer for at programmeringens nytteverdi må gjøres tydeligere for både lærere og elever, slik at holdningene til læringsverktøyet kan stimuleres.

Flere av aspektene som presenteres i forskningsartiklene over, beskrives også i boka «Programmering i skolen» av Haraldsrud et al. (2020). Boka har som mål å bidra til at lærere kan ta gode valg i undervisningen av programmering, og fremhever samtidig noen av de mulighetene og utfordringene som kan oppstå underveis. Ifølge forfatterne åpner programmering opp for å utforske kjente konsepter på en my måte, samtidig som det gir elevene muligheter til å prøve og feile. Programmeringsaktiviteter legger til rette for nye tilnærminger til fagstoffet, og kan gjøre store og avanserte oppgaver mindre og mer overkommelige å løse. På samme måte som Vinnervik (2021), hevder også Haraldsrud et al. (2020) aspektet med tid som en mulig utfordring, ved at programmering tar tid fra annen undervisning. Motivasjon er annen mulig utfordring, i form av at det for lærere kan være utfordrende å motivere både høytpresterende – og lavtpresenterende elever. Forfatterne understreker derfor, på samme måte som Munthe (2022) og Sengupta et al. (2013), viktigheten av å differensiere oppgaver slik at flest mulig kan oppleve mestring. En måte å oppnå dette på, er å utvikle opplegg der elevene får nok støtte og hjelp underveis, samtidig som det gir nok utfordringer til elever som kan mer. De erkjenner at det alltid vil

⁶ Deler av avsnitt er hentet fra tidligere selvskrevet oppgave i PPUT301.

være elever som ikke liker programmering, men at vi som undervisere likevel må "fortsette å strekke oss så langt som mulig slik at så mange som mulig føler mestring og glede" Haraldsrud et al. (2020, s. 163)⁷.

I tillegg til å understreke behovet for en biologiundervisning som holder følge med dagens biologi, presenterer Pevzner og Shamir (2009) i sin artikkel flere implikasjoner for undervisning av faget. Forfatterne presenterer det samme pedagogiske "dilemmaet", som beskrives flere steder i litteraturen, nemlig spørsmålet om *hvordan* man på best mulig måte kan integrere programmeringen i faget på en god måte. De anerkjenner utfordringene med å skape et slikt undervisningsopplegg, men presenterer likevel noen av erfaringene de har gjort seg i en tidligere gjennomført studie. Undervisningsopplegg bør, ifølge Pevzner og Shamir (2009), kreve *få* forkunnskaper hos studentene, fremheve verktøys nytteverdi i faget og gjerne bygge på en «historie» som spiller på biologistudentenes interesser for å skape engasjement⁸.

3. 2 Programmering i undervisningen – kjennetegn ved elevenes holdninger

Videre følger en presentasjon av et utvalg litteratur som mer spesifikt sier noe om elevenes *holdninger* til programmering.

I 2022 gjennomførte Forrester et al. (2022) en studie på amerikanske biologistudenters holdninger til å programmere i biologiundervisningen på universitetsnivå. Studiens utvalg besto av 14 studenter som deltok i et kurs med programmering i programmeringsspråket R. Innledningsvis i artikkelen viser forfatterne til deres motivasjon for studien, og presenterer tidligere studier som har funnet at flere studenter, og særlig kvinnelige, opplever angst («anxiety») og bekymring i møte med programmering i undervisning. I følge forfatterne kan denne programmeringsrelaterte angsten resultere i lavt faglig utbytte, lav mestringsfølelse og lavt engasjement i møte med verktøyet i fag (Forrester et al., 2022). I forbindelse med studien gjennomførte forfatterne et undervisningsopplegg der studentene samarbeidet med programmeringsoppgaver i en biologisk kontekst. Etter gjennomføringen konkluderte forskerne med at studentenes nivå av angst hadde gått ned, samtidig som elevene rapporterte høyere grad av selvoppfatning og kontroll i tilfeller der de programmerer i faget. Forfatterne argumenterer på bakgrunn av sine funn for at «lavterskel» undervisningsopplegg med hyppige tilbakemeldinger der studentene kan jobbe sammen i et aktivt læringsmiljø, kan ha en positiv effekt på studentenes holdninger til verktøyet. Det er også en fordel dersom studentene føler *eierskap* til det de holder på

⁷ Deler av avsnitt er hentet fra tidligere selvskrevet oppgave i PPUT301.

⁸ Deler av avsnitt er hentet fra tidligere selvskrevet oppgave i PPUT301.

med, og Forrester et al. (2022) foreslår at man kan la elevene bruke egne data i programmeringsaktiviteter. Videre bør lærere være tilgjengelige i læringsprosessene, slik at elevene kan stille spørsmål underveis og få støtte og veiledning i situasjoner der de er usikre. Forfatterne argumenterer avslutningsvis for at det bør gjennomføres videre forskning på «programmerings-angst» i biologifaget i skolen⁹.

Også i Norge ser vi tendensen av de samme holdningene hos biologistudenter ved høyere utdanning. Ved universitetet i Oslo ble det gjennomført en undersøkelse på biologistudenters holdninger til beregningsorientert biologi, i faget *BIOS1100 - Innføring i beregningsmodeller for biovitenskap*, der elevene må programmere og modellere som en del av faget. Eliassen et al. (2021) argumenterer for at innføringen av programmering som en del av andre fag, kan by på både muligheter og utfordringer. En mulighet handler om at verktøyene gir studentene mulighet til å jobbe med mer realistiske problemstillinger i sin utdanning. Likevel opplever mange studenter programmering som utfordrende, da flertallet har lite eller ingen programmeringserfaring fra tidligere. I sin studie ønsket forfattere å få "mer kunnskap om motivasjon og læring i emner som integrerer programmering som et nytt element i ellers veletablerte fagdisipliner" (Eliassen et al., 2021, s. 58). De forsøkte å kartlegge studentenes holdninger, med særlig fokus på *interesse og mestringsforventning*. Resultatene viste at studentene hadde noe lavere mestringsforventning og mye lavere interesse for emnet som inneholder programmering i biologi, sammenlignet med studieretningen generelt. Forfatterne hevder at dette kan gjøre studentene mer sårbare dersom de møter motgang underveis i semesteret, som igjen kan være uheldig for elevenes læringsprosesser. For å forsøke å minske denne utfordringen, argumenterer Eliassen et al. (2021) for at man bør videreutvikle nytteverdien studentene ser i emnet, og forsøke å øke studentenes interesse for oppgaver med programmering. Dette kan gjøres ved å fremheve programmeringens verdi og relevans i kommende emner og i arbeidslivet, gjerne ved at studentene selv får velge tema og problemstillinger. I tillegg kan programmeringen hektes på noe studentene allerede har interesse for, gjerne knyttet opp mot dagsaktuelle temaer¹⁰.

Funnene omkring studentenes holdninger presentert i artiklene ovenfor, finner vi igjen i flere andre studier. Baser (2013) fant i sin studie at det er en stor sammenheng mellom studentenes opplevelse av mestring når de programmerer, og holdningene de har til verktøyet. Studien ble gjennomført på et tyrkisk universitet, der utvalget var førsteårsstudenter i et innføringskurs i programmering (n = 179). Forfatteren viser innledningsvis til litteraturen som hevder at studenter kan oppleve programmering

⁹ Deler av avsnitt er hentet fra tidligere selvskrevet oppgave i PPUT301. Pevzner, P. & Shamir, R. (2009). Computing has changed biology—biology education must catch up. *Science*, 325(5940), 541-542.

¹⁰ Deler av avsnitt er hentet fra tidligere selvskrevet oppgave i PPUT301.

som vanskelig, usosialt og kjedelig (Farkas & Murthy, 2005, via Baser, 2013). Dette kan resultere i at studentene ofte har negative holdninger til det å programmere, som igjen kan påvirke elevenes prestasjon i møte med kompetansen. Funn fra studien viser at kvinnene jevnt over hadde flere negative holdninger til programmering, sammenlignet med menn. Forfatteren argumenterer for at økt mestring i møte med kompetansen kan oppnås dersom studentenes holdninger stimuleres, og fremhever viktigheten av at lærere tar hensyn til disse holdningene ved valg av metoder og design for undervisning med programmering ¹¹.

Erol (2020) utforsker i sin studie, virkningen av et spesifikt undervisningsopplegg på studenters holdninger og mestringsevne i møte med programmering. Forfatteren viser innledningsvis til flere utfordringer ved implementeringen av programmering i undervisningen, og fremhever betydningen av elevenes holdninger og mestringsevne på deres prestasjoner. Studien til Erol (2020) ble gjennomført på en gruppe studenter ved et data-teknisk studie i Tyrkia (n = 25), der studentene skulle gjennomføre et 12 uker langt undervisningsopplegg med aktiviteter i robotdesign og programmering. Studiens formål var å undersøke virkningen av opplegget på studentenes holdninger og mestringsevne til programmering, og å identifisere de motiverende og utfordrende («hemmende») faktorene i elevenes opplevelser. Resultatene fra studien viste at det var en signifikant forskjell i elevenes rapporterte selvtillit og motivasjon, før og etter gjennomføringen av undervisningsopplegget. Aktiv læring, eierskap og nysgjerrighet var blant faktorene som elevene opplevde som motiverende, mens tekniske problemer og manglende tidligere erfaringer ble beskrevet som noen av de utfordrende faktorene. En av de viktigste konklusjonene som gjøres med bakgrunn i studien, er ideen om at elevenes holdninger kan påvirkes, basert på valg av undervisningsopplegg. Noen av effektene på elevenes holdninger var økt engasjement, økt opplevd mestring og økt interesse for programmering.

På lignende vis som Erol (2020), utforsket Du et al. (2016) i deres studie, virkningen av et undervisningsopplegg på elevenes holdninger og ferdigheter i programmering. Forfatterne argumenterer for at programmering er en utfordrende ferdighet å både lære seg og undervise, og at flere studenter uttrykker lav interesse for verktøyet. Du et al. (2016) presenterer et lignende problem som hos Forrester et al. (2022), nemlig at programmering kan føre til angst («anxiety») blant elever som lærer seg det (Raub, 1981 via (Du et al., 2016)). På bakgrunn av utfordringene ønsker forfatterne å undersøke om undervisningsopplegget «Hour of Code» kan forbedre elevenes holdninger og forståelse av programmering. Undervisningsopplegget baserer seg på interaktiv læring hvor elevene skal samarbeide for å løse «spill lignende» oppgaver med programmering. Studien ble gjennomført på en

¹¹ Deler av avsnitt er hentet fra tidligere selvskrevet oppgave i PPUT301.

gruppe amerikanske bachelorstudenter, hvor flertallet hadde lite eller ingen tidligere erfaringer med programmering. For å måle effekten av undervisningsopplegget på elevenes holdninger og ferdigheter, besvarte studentene to spørreskjema – et før og et etter gjennomføringen av undervisningsopplegget. Resultatene viste at flesteparten av elevene hadde positive opplevelser med undervisningsopplegget, og at opplegget hadde en positiv effekt på studentenes holdninger ved at det økte bevisstheten omkring viktigheten av programmering. Imidlertid ble det *ikke* funnet en signifikant forskjell på elevenes programmeringsferdigheter. Derfor antyder forfatterne at undervisningsopplegget ikke nødvendigvis vil forbedre studentenes ferdigheter innen programmering alene, men at en kombinasjon av «Hour of Code» og mer tradisjonell undervisning i programmering kan være nødvendig for å oppnå en slik endring.

Det er ennå begrenset med forskning på elevers holdninger til programmering på videregående skoler i Norge. Derfor har jeg valgt å inkludere to masteroppgaver som ser på holdninger til programmering inn i matematikkfaget. Masteroppgaven til Zukanovic (2021) har til hensikt å undersøke hva som kjennetegner matematikklæreres holdninger til undervisning av programmering i matematikk, og forfatteren fant at lærerne tenderte mot en ambivalent holdning til det å undervise programmering i faget. Holdningene til lærerne var preget av utfordringer som manglende kompetanse blant både lærere og elever, ny eksamensform og tekniske problemer. Til tross for at disse utfordringene kunne virke frustrerende på både lærere og elever, så mange av lærerne verdien av kompetansen i faget.

Dette inntrykket støttes av funnene i masteroppgaven til Andersen (2022) sin masteroppgave, der *elevenes* holdninger til programmering ble undersøkt. Blant sine funn fant hun en tydelig sammenheng mellom grad av mestring på oppgaver, og følelsene elevene får til programmering i matematikk. Denne sammenhengen finner støtte i litteraturen (Baser, 2013; Eliassen et al., 2021; Forrester et al. 2022). I tillegg fant forfatteren at mange av elevene beskriver programmering som «vanskelig», og noe som gjør selve faget vanskeligere. Flere av informantene i klassen forteller at de ikke liker programmering fordi de ikke kan nok. Jevnt over rapporterte elevene lavere motivasjon til programmering, sammenlignet med den tradisjonelle matematikken. Avslutningsvis argumenterer Andersen (2022) for at disse utfordringene kan minskes dersom elevene opplever mestring, og ved at de får nok tid til å utforske læringsverktøyet uten å føle at de blir vurdert på hvor mye de kan. Ifølge Andersen (2022) må elevene få meningsfulle oppgaver og muligheten til å utforske hva programmering er og hvordan det kan brukes i den virkelige verden ¹².

¹² Deler av avsnitt er hentet fra tidligere selvskrivet oppgave i PPUT301.

Kapittel 4: Metode

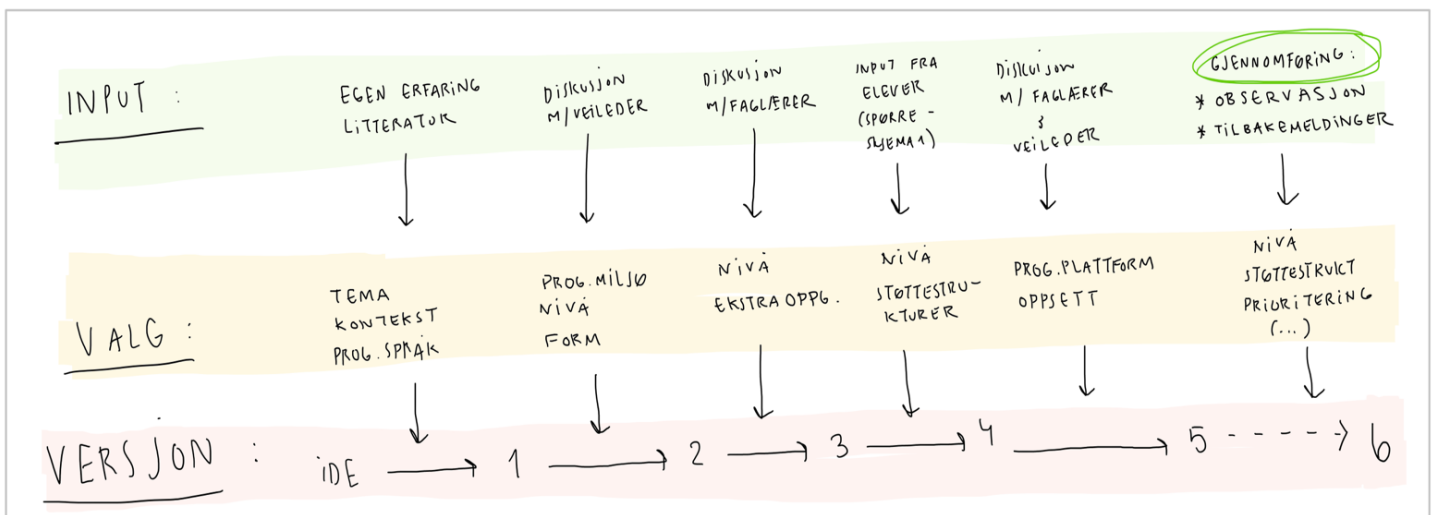
I følgende kapittel vil jeg redegjøre for studiens metode. Først vil jeg presentere undervisningsopplegget, «Code to Cure», som jeg har utviklet i forbindelse med studien. Videre vil metodevalg for oppgavens datainnsamling og dataanalyse presenteres, før jeg avslutningsvis vil drøfte studiens validitet og reliabilitet.

4. 1 Undervisningsopplegget «Code to Cure»

Undervisningsopplegget «Code to Cure» ble utviklet i perioden september 2022 til januar 2023. I dette delkapitlet vil jeg beskrive prosessen, samt de valgene som ble tatt i forbindelse med utviklingen av «Code to Cure». Videre vil jeg beskrive gjennomføringen og presentere selve undervisningsopplegget.

4. 1. 1 Prosessen med utviklingen av undervisningsopplegget

Proessen startet med en ide og et ønske om å lage et undervisningsopplegg som kunne gi biologi 2 elever et positivt møte med programmering i faget. Valgene er tatt med utgangspunkt i innspill fra elevene, egen erfaring, litteratur og diskusjon med faglærer og veileder. I figuren nedenfor (figur 5) er de ulike iterasjoner med tilhørende input og valg, presentert.



Figur 5: Figur som viser utviklingen av undervisningsopplegget "Code to Cure"

Jeg bestemte meg tidlig for å inkludere veileder, faglærer og elevene i prosessen med å utvikle opplegget. Hovedveileder Morten Munthe har erfaring med å inkludere programmering inn i skolefaget matematikk, og i våre diskusjoner fikk jeg verdifulle innspill på hvordan programmering kan brukes for

faget, og ikke omvendt (Munthe, 2022). Faglærer har en sterk biologifaglig kompetanse, og kjenner i tillegg elevene som skal delta i prosjektet og deres forutsetninger. Det var viktig for meg at faglærer fikk eierskap til prosjektet og en følelse av å bli inkludert i prosessen. Andre verdifulle innspill fikk jeg fra elevene som skulle delta i prosjektet. I spørreskjemaene fikk jeg informasjon om elevenes holdninger og deres refleksjoner omkring kjennetegn ved et godt undervisningsopplegg med programmering.

På bakgrunn av egne erfaringer og eksisterende teori, valgte jeg en *kontekstbasert tilnærming* som utgangspunkt for mitt undervisningsopplegg. Kontekstbasert undervisning kan i følge Nergård (2015) bidra til positive holdninger hos elever, ved at den synliggjør fagets kobling til livet utenfor skolen. Forhåpentligvis ville en autentisk problemstilling bidra til at elevene opplevde verktøyet som relevant og meningsfullt (Nergård, 2015). I litteraturen finnes det støtte for at opplevd relevans kan føre til økt dybdelæring (Munthe, 2022; Sengupta et al., 2013; Waters, 2020; Øyehaug, 2019), økt interesse (Munthe, 2022; Pevzner & Shamir, 2009; Sengupta et al., 2013) og positive holdninger til programmering i faget (Andersen, 2022; Eliassen et al., 2021). Tidligere studier har vist at kontekstbasert undervisning er positivt også når elevene skal *programmere* (Nergård, 2015; Pevzner & Shamir, 2009; Sengupta et al., 2013).

Valg av programmeringsspråk i undervisningsopplegget ble Python. Python er det programmeringsspråket som brukes på videregående skoler i dag (Munthe, 2022), og Ozgur et al. (2017) beskriver Python som et godt egnet programmeringsspråk i undervisningssituasjoner. Språket er enkelt å bruke, samtidig som det er svært populært og allsidig programmeringsspråk – også innen biologi (David, 2021). Sammen med hovedveileder drøftet jeg fordeler og ulemper ved flere forskjellige plattformer. Tidlig i prosessen var opplegget lagt til Jupyter Notebook¹³ – en interaktiv, web-basert plattform, spesielt egnet for undervisning med programmering. Til tross for flere fordeler ved Jupyter Notebook, falt valget til slutt på å lage en egen nettside med interaktive programmeringsoppgaver innfelt i selve nettsiden. Nettsiden ble designet i Google Sites, og de interaktive programmeringsoppgavene ble levert av Trinket.io¹⁴. Nettsiden utgjorde et interaktivt miljø, der elevene skulle forflytte seg mellom forskjellige «rom» på nettsiden og løse programmeringsoppgavene de møtte på veien. Slik ble elevene kun presentert for én oppgave av gangen, som forhåpentligvis kunne dette bidra til at den samlede belastningen på elevene ble mindre (Guzman et al., 2019). Ved å integrere programmeringsoppgavene direkte i nettsiden, kunne elevene skrive og kjøre kode på én og samme plattform. En svakhet ved denne løsningen, er at elevene ikke fikk mulighet til å lagre koden sin

¹³ <https://jupyter.org>

¹⁴ <https://trinket.io>

underveis. Jeg valgte å løse dette ved å gi elevene instruksjoner om å åpne trinket-editorene i en egen fane på datamaskinen, og å ta skjermbilder av koden underveis.

Elevenes forutsetninger sto svært sentralt i utviklingsprosessen av undervisningsopplegget. I datainnsamlingen inkluderte jeg spørsmål for å kartlegge elevenes erfaringer med programmering. Resultatene indikerte at flere av elevene hadde lav mestringfølelse, få forkunnskaper og en oppfatning av at programmering var utfordrende og vanskelig (se resultatkapittel – RQ1).

I samråd med veileder og faglærer bestemte jeg meg på bakgrunn av dette for å lage et undervisningsopplegg som krevde få forkunnskaper innen programmering, og jeg valgte å legge programmeringen på et mer grunnleggende nivå slik at flest mulig kunne oppleve mestring (Forrester et al., 2022; Munthe, 2022; Sengupta et al., 2013).

Basert på anbefalinger fra litteraturen ble lagt opp til at elevene skulle jobbe *sammen* i programmeringsaktiviteten (Du et al., 2016; Forrester et al., 2022; Munthe, 2022). Munthe (2022) anbefaler å legge programmeringsaktiviteter til et tema som allerede er kjent for elevene, og på bakgrunn av dette valgte jeg å legge gjennomføringen til *etter* elevene hadde gjennomgått kapitlet med faglærer.

4. 1. 2 Beskrivelsen av gjennomføringen og undervisningsopplegget «Code to Cure»

Her vil jeg beskrive undervisningsopplegget, «Code to Cure», og gi en kort oversikt over gjennomføringen av økta. Merk at undervisningsopplegget som beskrives her, er det *gjennomførte* undervisningsopplegget (versjon 5 i figur 5).



Illustrasjon laget av meg til undervisningsopplegget

I undervisningsopplegget «Code to Cure», skal elevene gå inn i rollen som leger, og i løpet av et to-delt opplegg både *diagnostisere* og *behandle* en fiktiv pasient. Todelingen gjorde det mulig å gjennomføre

opplegget fordelt på to forskjellige undervisningsøkter. I mitt tilfelle ble de to delene gjennomført sammenhengende på en fagdag med tre skoletimer til disposisjon. Elevene jobbet sammen i grupper på 2-3 elever. Tabell 1 viser en oversikt over de ulike delene av gjennomføringen og anslagsvis tidsbruk på hver aktivitet.

Tabell 1: En oversikt over tidsramme og komponentene i undervisningsopplegget.

Tidsramme (anslagsvis)	Aktivitet
10 min	Informasjon om opplegget. Elevene får utdelt pasient, læringssti og tilgang til nettside
45 min	Elevene jobber med Del 1: Diagnostikk av pasient
45 min	Elevene jobber med Del 2: Behandling av pasient/ekstraoppgave

Elevene fikk i starten av økta utdelt et fysisk hefte med en læringssti (vedlegg B) og skulle løse oppgavene på en interaktiv nettside med forskjellige rom: venterom, laboratoriet og kontor. Læringsstien fortalte elevene hva de skal gjøre, og hvor på nettsiden de fant neste oppgave. Alle programmeringsoppgavene skulle gjøres direkte i trinket-editoren på nettsiden. Utdrag fra undervisningsopplegget med link ligger vedlagt, se vedlegg C.

Del 1: Diagnostikk av pasient

Hver gruppe fikk utdelt én av tre pasienter på venterommet de sammen skulle diagnostisere. For å komme frem til riktig diagnose måtte elevene undersøke pasientens symptomer og sykdomshistorie. Som siste steg i diagnostikken må elevene genteste pasienten for tre mulige sykdommer. Her ble elevene introdusert for oppgaven «den skeptiske legen», der én av elevene ved hjelp av programmering ble valgt tilfeldig til å gjennomføre en utfordring. Legen stoler ikke på at datamaskinen ved hjelp av programmering kan finne ut av hvilket sykdoms-gen som gjemmer seg i pasientens DNA. Skeptikeren skal lete etter en bestemt gensekvens på 4 basepar, gjemt i liten del av en DNA-sekvens på totalt 100 000 basepar. Elevene skulle i dette eksperimentet undersøke hvor lang tid de ville brukt på å lete gjennom *hele* gen-sekvensen manuelt. Dette resultatet skulle til slutt overbevise skeptikeren om at programmering var en raskere og mer nøyaktig metode. Ved den første programmeringsoppgaven skulle elevene lage en kodesnutt som ba datamaskinen lete etter en bestemt gen-sekvens som forårsaket sykdom, for eksempel sekvensen "GCTGGAT", i pasientens DNA. Elevene skulle teste hver av de tre mulige sykdommene, der kun én av sykdommene skulle gi treff i DNA-sekvensen. Avslutningsvis måtte elevene sende rapport til overlegen med sine funn via et forms-skjema som ligger integrert i nettsiden, og elevene fikk tilbakemelding på om de har funnet riktig sykdom.

Del 2: Behandling av pasient

Videre skulle elevene *behandle* pasienten ved hjelp av genterapi. Opplegget inkluderte informasjon om CRISPR, og hvordan metoden kan simuleres ved hjelp av programmering i Python. Elevene gjorde først noen enklere eksempler hvor de skulle finne og erstatte deler av en sekvens ved hjelp av programmering. Så fikk elevene i oppgave å lage en kodesnutt som 1) bekreftet at pasienten hadde sykdoms-gen i sitt DNA, 2) brukte det de hadde lært til å finne og erstatte dette genet med et friskt gen og 3) sjekket at sekvensen faktisk var borte, og skrive ut en beskjed om at behandlingen var vellykket. Avslutningsvis skulle elevene skrive en epikrise med informasjon om diagnostikk og behandling, formulert på et språk pasienten kunne forstå.

Som en ekstraoppgave til elevene som blir ferdig før tiden, ble det på nettsiden lagt inn to ekstra pasienter på venterommet: Anna og Jonas. Her skulle elevene gjennom flere steg lage et mer avansert program som simulerte kryssninger og resultater fra kryssningsskjemaer elevene vanligvis gjør for hånd. Informasjonen de fikk ved å bruke programmet, ga elevene et utgangspunkt når de skulle gi Anna og Jonas genetisk rådgiving om sykdom.

4. 2 Metode og forskningsdesign

Formålet med denne studien er å undersøke hvordan undervisningsopplegget «Code to Cure», kan påvirke biologi 2 elevs holdninger til programmering i faget. For å si noe om en eventuell effekt av undervisningsopplegget, er det avgjørende å kartlegge hvilke holdninger elevene faktisk har til det å skulle programmere i faget, og hvilke faktorer som er av betydning når disse holdningene formes. Den unike konteksten i min studie er den aktuelle biologi 2 klassen der elevene har lite eller ingen erfaring med programmering fra tidligere. For å belyse oppgavens forskningsspørsmål, falt valg av forskningsdesign på *enkeltcase-studien*, som er en foretrukket strategi når målet med studien er en grundig og dypere forståelse for én enkeltcase (Postholm & Jacobsen, 2018).

I klasseromsforskning der de fenomenene som undersøkes ofte er komplekse og bør belyses fra forskjellige perspektiver, kan *mixed methods* som forskningsdesign være hensiktsmessig, skriver Brevik og Mathé (2021). Mixed methods er et forskningsdesign der komplekse fenomener belyses gjennom en kombinasjon av kvantitative og kvalitative data (Brevik & Mathé, 2021). I denne studien utgjør elevenes svar på de lukkede spørsmålene fra spørreskjemaene det kvantitative datamaterialet. De åpne spørsmålene fra spørreskjemaene, utgjør sammen med transkripsjonene fra fokusgruppeintervjuene,

det kvalitative datagrunnlaget. Tabell 2 viser en oversikt de kvalitative og kvantitative datakildene i dette forskningsprosjektet.

Tabell 2: En oversikt over oppgavens datakilder.

Design	Kvalitative datakilder	Kvantitative datakilder
Mixed methods	Spørreskjema 1 (åpne spørsmål)	Spørreskjema 1 (lukkede spørsmål)
	Spørreskjema 2 (åpne spørsmål)	Spørreskjema 2 (lukkede spørsmål)
	Fokusgruppeintervju 1	Spørreskjema 3 (lukkede spørsmål)
	Fokusgruppeintervju 2	
	(Observasjonsnotater)	

Til tross for at jeg i min studie vil ha en vekting mot kvalitative datakilder, vil en slik mixed-methods tilnærming forhåpentligvis kunne bidra til å skape et mer helhetlig bilde og en økt forståelse for det fenomenet som undersøkes (Brevik & Mathé, 2021). Det kvantitative datamaterialet kan gi et bilde av hva elevene i klassen mener mer «samlet sett», og samtidig gi en pekepinn på *om og i hvilken grad* informantene i intervjuene kan sies å representere opplevelsene til også resten av elevene i klassen. Det kvalitative datamaterialet kan bidra inn med en dypere innsikt i enkelte av elevenes opplevelser – beskrevet med deres egne ord. Denne kombinasjonen av metode finner jeg støtte for i metodelitteraturen (Postholm & Jacobsen, 2018).

4. 2. 1 Studiens utvalg

Studiens deltakere omfatter 28 elever i alderen 18-19 år i en biologi 2 klasse ved en videregående skole på Østlandet. Skolen har forholdsvis høyt karaktersnitt ved inntak, og faglærer opplever den aktuelle biologiklassen jevnt over faglig sterke og motiverte for faget. Elevene har fulgt ny læreplan siden 2020, og i år (2023) er første året hvor vg3 følger ny læreplan etter fagfornyelsen. Elevene har programmert én gang tidligere i faget, men flere har erfaringer fra andre fag som matematikk, fysikk og kjemi (se resultatkapittel RQ1). Faglærers inntrykk er at mange av elevene har få forkunnskaper og negative holdninger til å programmere generelt og i faget. Dette inntrykket støttes i kartleggingen av elevenes holdninger i pre-undersøkelsen (se resultatkapittel RQ2).

Skolen er min tidligere arbeidsplass, og utvalget ble delvis valgt på bakgrunn av tilgjengelighet.

På bakgrunn av dette, ble utvalget på mange måter et *bequemmelighetsutvalg* (Blikstad-Balas & Dalland, 2021). Det var viktig for meg at faglærer følte seg inkludert i prosessen med utviklingen av undervisningsopplegget, og det at jeg og faglærer kjente hverandre fra tidligere gjorde det enkelt å spørre om faglærers mening og innspill underveis.

Alle elevene i klassen fikk invitasjon til å delta i prosjektet. På samtykkeskjemaet fikk elevene mulighet til å huke av for om de ønsket å være med på spørreskjemaundersøkelsen, fokusgruppeintervjuet eller begge deler. Se vedlegg D for samtykkeskjema elevene fikk utdelt. Da gjennomføringen av datainnsamlingen ble satt til tre ulike dager, varierer utvalgets størrelse ved pre- og post-spørreskjema grunnet fravær blant elevene. Tabell 3 viser en oversikt over antall elever og kjønnsfordeling i studiens ulike utvalg.

Tabell 3: Tabell som viser en oversikt over antall og kjønnsfordeling i studiens utvalg.

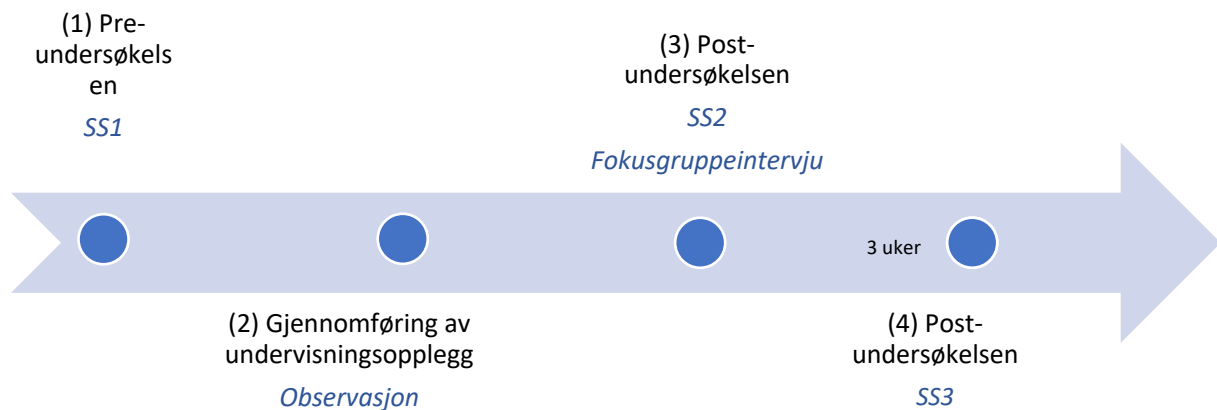
Datakilde	Antall	Jenter	Gutter
Pre-undersøkelsen (SS1)	21	15	6
Post-undersøkelsen (SS2)	28	18	10
Post-undersøkelsen (SS3)	21	15	6
Fokusgruppeintervju 1	3	3	0
Fokusgruppeintervju 2	3	1	2

I utgangspunktet var tanken at informantene til fokusgruppeintervjuet skulle settes sammen basert på svarene de leverte i post-spørreskjema slik at jeg kunne selektere elever med ulike forkunnskaper innen programmering og ulik opplevelse av undervisningsopplegget (Tjora, 2021). Da kun seks elever samtykket til å delta i intervju, ble det valgt å gå bort fra dette kriteriet. Det er derfor ikke stilt krav om spredning i forkunnskaper og opplevelser til undervisningsopplegg når informanter til fokusgruppeintervju er valgt.

4. 2. 2 Datainnsamlingsmetoder i studien

I denne delen av metodekapitlet vil jeg presentere de metodiske valg og vurderinger som er gjort for å belyse studiens hensikt og forskningsspørsmål. Datamaterialet i studien er dannet på bakgrunn av i hovedsak *to hovedkomponenter*, kalt pre-undersøkelsen og post-undersøkelsen. Pre-undersøkelsen beskriver datainnsamlingen i form av en spørreundersøkelse som ble gjennomført *før* elevene hadde gjennomført «Code to Cure», videre referert til som SS1 (= spørreskjema 1). Rett etter gjennomføringen besvarte elevene et nytt spørreskjema, SS2 (= spørreskjema 2), og 6 elever deltok i to fokusgruppeintervjuer. Tre uker etter gjennomføringen av undervisningsopplegget besvarte elevene SS3 (=spørreskjema 3). Samlet utgjør SS2, SS3 og fokusgruppeintervjuene *post-undersøkelsen*.

I tillegg til disse hovedkomponentene, skrev både jeg og faglærer ned observasjoner fra selve gjennomføringen av opplegget. Figur 6 viser en oversikt over komponentene i mitt datamateriale.



Figur 6: Figur som viser en oversikt over datamaterialet i studien.

Kvantitativ metode: Spørreundersøkelsene

Spørreundersøkelser ble valgt som datainnsamlingsmetode i både pre- og post-undersøkelsen. Elevene besvarte tre spørreskjema – SS1 i pre-undersøkelsen, og SS2 og SS3 i post-undersøkelsen, der hensikten var å få innsikt i elevenes holdninger til å programmere i biologi 2, samt få informasjon om opplevelser og erfaringer med undervisningsopplegget «Code to Cure». Frønes og Pettersen (2021) argumenterer for at spørreundersøkelse som metode er hensiktsmessig i tilfellene der ønsket er å få informasjon om, beskrive, sammenligne og forklare fenomener som holdninger, slik det er i denne studien. Videre argumenterer forfatterne for at metoden kan være egnet for å måle effekt – for eksempel effekten av et bestemt undervisningsopplegg. Spørreskjemaene åpnet opp for at flere elever kunne delta i studien, slik at flere synspunkter ble representert i studiens datamateriale.

Spørreskjemaene ble en kombinasjon av åpne tekstsvar, avkrysning og påstander elevene skulle gradere ved hjelp av en fem-steps Likert-skala, fra «helt uenig» (1) til «helt enig» (5). Valget om å inkludere en slik gradering av påstander ble tatt med bakgrunn i metodelitteraturen, som hevder at formen kan være en god måte å måle komplekse fenomener som holdninger og følelser (Baser, 2013; Frønes & Pettersen, 2021). Det ble inkludert en kombinasjon av positive og negative utsagn, slik at retningen på spørsmålene varierte (Brevik & Mathé, 2021). I tillegg ble det inkludert en midtkategori kalt «nøytral» slik at elevene som faktisk hadde en slik holdning, ikke ble nødt til å si seg enig eller uenig i påstanden (Brevik & Mathé, 2021; Postholm & Jacobsen, 2018). Både avkrysningsspørsmålene og påstandene bidro til at utfyllingen av spørreskjemaene tok mindre tid for elevene, der målet var å lage en så kort og presis

spørreundersøkelse som mulig, av respekt for respondentenes tid (Brevik & Mathé, 2021). Formen var i tillegg hensiktsmessig for den senere behandlingen av datamaterialet, ved at svar på disse spørsmålene forholdsvis «enkelt» kunne gjøres om til tallverdier for å: 1) vise fordelingen av elevenes svar, og 2) gi et sammenligningsgrunnlag for å sammenligne elevenes svar på pre- og post-undersøkelsen (Brevik & Mathé, 2021). De åpne tekstsvarene åpnet opp for flere innfallsvinkler og ga elevene rom til å reflektere utover de lukkede avkrysningsoppgavene og påstandene (Brevik & Mathé, 2021). Samlet utgjorde spørreundersøkelsens struktur en mixed-method tilnærming – der kvantitative data i form av avkryssing og graderinger, ble supplert med kvalitative data fra de åpne tekstoppgavene (Brevik & Mathé, 2021). Spørsmålene og påstandene er valgt og formulert av meg, inspirert av funn og spørreskjema fra tidligere forskningsprosjekter (Tapia & Marsh, 2002; Yukselturk & Altiok, 2017) og hadde et gjennomgående fokus på aspektene mestringsfølelse, engasjement og nytteverdi.

Spørreundersøkelsene var ikke anonyme, på bakgrunn av at jeg opprinnelig ønsket å invitere enkelte elever til et fokusgruppeintervju i etterkant av undervisningsopplegget. Til tross for at jeg gikk vekk fra en slik selektering, valgte jeg likevel å be om navn på alle skjemaene, slik at jeg hadde muligheten til å sammenligne svarene til den enkelte elev på pre- og post-undersøkelsen. Det digitale nettskjemaet ved UiO (Nettskjema, u.å) ble brukt for digital besvarelsene av spørreskjema.

Spørreskjema 1 ble besvart av elevene i pre-undersøkelsen, der formålet med undersøkelsen var å 1) bli bedre kjent med utvalget, 2) kartlegge elevenes holdninger til programmering, og 3) få innspill til revideringen av undervisningsopplegget. Det ble valgt å inkludere spørsmål om kjønn, programmeringsbakgrunn, matematikkbakgrunn, motivasjon for å velge biologi som valgfag og interesser. Disse opplysningene var med på å beskrive utvalget og skape et tydeligere bilde av hvem elevene var. For å kartlegge elevenes holdninger til programmering, valgte jeg å dele spørreskjemaet i to hoveddeler – en generell del og en del spisset inn mot biologifaget. Avslutningsvis fikk elevene to åpne spørsmål der de ble bedt reflektere omkring hva som kjennetegner et godt undervisningsopplegg i biologi. Se spørreundersøkelsen vedlagt (vedlegg E). Undersøkelsen fant sted i desember 2022, der 21 av de 22 elevene som var til stede deltok i undersøkelsen. En hovedvekt av datamaterialet som ble samlet inn ble brukt til å besvare RQ1 og RQ2. Elevenes refleksjoner omkring undervisningsopplegg med programmering ble viktig i prosessen med å utvikle undervisningsopplegget.

Et av formålene med *spørreskjema 2* var å kartlegge elevenes holdninger til programmering i biologi etter gjennomføringen av undervisningsopplegget, som et utgangspunkt for å sammenligne elevenes svar fra pre-undersøkelsen (Frønes & Pettersen, 2021). Spørreskjemaets andre formål var å få innsikt i elevenes opplevelser med undervisningsopplegget «Code to Cure». Post-undersøkelsen tok sted i en

biologitime i januar - etter gjennomføringen av undervisningsopplegget. Alle elevene i klassen fikk mulighet til å delta, også de elevene som var ikke var til stede ved pre-undersøkelsen. Av elevene som var til stede samtykket 28 av 28 elever til å delta i forskningsprosjektet. SS2 hadde samme mixed-methods tilnærming som SS1 (Brevik & Mathé, 2021), med en kombinasjon av avkrysning, påstander og åpne tekstsvar. Påstandene om *programmering generelt i undervisningen* fra SS1, ble inkludert også i denne undersøkelsen, for å kunne sammenligne elevenes graderinger før og etter gjennomføringen. Datamaterialet fra disse spørsmålene ville i hovedsak bli brukt til å belyse RQ3. For å få innsikt i elevenes opplevelse og erfaringer med selve undervisningsopplegget, inkluderte jeg påstander, samt åpne tekstsvar der elevene kunne svare mer utfyllende om opplegget. Spørsmål fra SS2 ligger vedlagt som vedlegg F. En hovedvekt av datamaterialet fra SS2 ble brukt for å belyse RQ2 og RQ3.

Spørreskjema 3 ble sendt ut til elevene tre uker etter gjennomføringen av undervisningsopplegget. Formålet med undersøkelsen var å kartlegge elevenes holdninger til å programmere i faget etter at det var gått tid fra selve gjennomføringen. SS3 inneholdt de påstandene fra SS1 som handlet om programmering *spesifikt i biologi (5 påstander)*. Undersøkelsen ble sendt ut av faglærer, uten meg til stede i klasserommet, og 21 av elevene var til stede for å besvare dette spørreskjemaet. Elever hadde ingen annen undervisning med programmeringen i faget i perioden mellom gjennomføringen og SS3. Spørsmål fra SS3 ligger vedlagt som vedlegg G. Elevenes besvarelser ble brukt for å belyse RQ3.

Kvalitativ metode: Fokusgruppeintervju

Som en del av post-undersøkelsen deltok et utvalg av elevene i to forskjellige *semistrukturerte mini-fokusgruppeintervjuer*. I metodelitteraturen hevdes det at intervju som metode er hensiktsmessig når man som forsker ønsker større innsikt i menneskers perspektiver, erfaringer, følelser og tanker (Dalen, 2011; Kvale & Brinkmann, 2015; Postholm & Jacobsen, 2018). Intervjuet som metode åpnet opp for rike beskrivelser (Svenkerud, 2021), og ga meg som forsker mulighet til å få en dypere innsikt i elevenes holdninger til programmering og hva som former disse. Videre argumenterer Svenkerud (2021) for at intervju som metode gir elevene mulighet til å beskrive opplevelser og perspektiver med *egne ord*, og på den måten bringe frem flere innfallsvinkler enn de som ville kommet frem gjennom spørreundersøkelsene alene. Brevik og Mathé (2021) hevder at intervjuet kan bidra til en dypere forståelse, og å skape et mer nyansert bilde av det fenomenet som ble undersøkt.

Valg om å gjennomføre et *semistrukturert* intervju, ble tatt på bakgrunn av anbefalinger i metodelitteraturen, der Postholm og Jacobsen (2018) argumenterer for at intervjuformen er hensiktsmessig når formålet er å *forstå deltakernes perspektiver*. Det semistrukturerte intervjuet var

særlig hensiktsmessig ved at det åpnet opp for å ønske nye tema velkommen, samtidig som det på forhånd var avklart tema og hovedspørsmål (Postholm & Jacobsen, 2018). Da det finnes begrenset med tidligere litteratur som sier noe om videregående elevers holdninger til programmering i Norge, særlig i faget biologi, var det stor sannsynlighet for at det underveis i samtalen kunne dukke opp nye, interessante perspektiver og tema det var verdt å utforske videre.

Som intervjuform valgte jeg *fokusgruppeintervjuet*. Fokusgruppeintervjuet er en form for gruppeintervju der flere informanter samles for å diskutere et bestemt tema (Tjora, 2021), slik at de kan de sammenligne ulike erfaringer og forståelser (Kvale & Brinkmann, 2015). Deltakerne kan utfordre hverandre og trekke frem andre synspunkter enn det som er planlagt – slik at spontane reaksjoner og synspunkter kan bringes frem (Kvale & Brinkmann, 2015). Jeg vil argumentere for at denne intervjuformen er hensiktsmessig i min studie, da elevene kan diskutere og bygge videre på ulike synspunkter og erfaringer i gruppa – både generelle holdninger til å programmere i faget, men også opplevelsen de hadde med undervisningsopplegget. Kvale og Brinkmann (2015) presenterer en annen fordel med fokusgruppeintervjuet – nemlig at informantene kan oppleve det som en trygghet å sitte samlet i en gruppe i intervjusituasjonen. Videre argumenterer forfatterne for at intervjuformen er velegnet til studier der fenomenet som undersøkes er «nytt». Det er blitt gjort lite forskning på programmering i biologiundervisningen fra tidligere, så denne egenskapet ved fokusgruppeintervjuet kan være verdifullt i min studie for å bringe frem nye, interessante synspunkter.

Krueger (1994) foreslår det han kaller *mini-fokusgrupper*, fokusgruppeintervjuer med færre deltakere i hver gruppe. Tjora (2021) beskriver hvordan mindre fokusgrupper kan bidra til at deltakerne opplever økt trygghet, i tillegg til å minske sannsynligheten for presentasjonsangst for å delta i samtalen. I mitt tilfelle hadde seks elever svart at de ønsket å delta i intervjuet, og jeg valgte i samråd med faglærer å fordele elevene på to forskjellige mini-fokusgrupper. Jeg ønsket at intervjuet skulle oppleves så trygt som mulig for elevene, og kanskje ville det være enklere å oppnå dette dersom elevene var færre på hver gruppe (Tjora, 2021). Faglærer kjenner elevene, og mente dette kunne være en god løsning for å sikre at alle i gruppa turte å ta ordet.

En sentral del av planleggingen til det semistrukturerte intervjuet, ligger i *utarbeidingen av intervjuguiden* (Dalen, 2011; Svenkerud, 2021; Postholm & Jacobsen, 2018). Med kjennskap til teori og konkretisering av oppgavens hensikt i forskningsspørsmål, ble det formulert *fire hovedspørsmål* som skulle danne utgangspunkt for det semistrukturerte intervjuet. Det første spørsmålet omhandler elevenes holdninger til å programmere generelt, og ble stilt i et forsøk på å belyse RQ1. Jeg valgte å formulere dette spørsmålet som et indirekte spørsmål, der elevene kunne reflektere omkring elevers

lave engasjement, lave motivasjon og lave mestringsfølelse i møte med programmering i skolen. Jeg håpet at den indirekte formen ville gi verdifull informasjon om elevenes egen situasjon og holdninger, samtidig som elevene ble nødt til å sette seg inn i andre perspektiver enn sine egne. Videre ble elevene spurt om sine erfaringer med «Code to Cure», og om og i så fall *hvordan* undervisningsøkta hadde forandret deres syn på programmering i biologi. Helt til slutt diskuterte elevene kjennetegn ved et godt undervisningsopplegg med programmering i biologi. Jeg forberedte flere forslag til underspørsmål til hver av de fire hovedspørsmålene på min egen intervjuguide for å få respondentene til å videre utdype og beskrive sine erfaringer og tanker underveis i gjennomføringen (Kvale & Brinkmann, 2015; Postholm & Jacobsen, 2018). Intervjuguiden er vedlagt som vedlegg H.

Alle de seks elevene som hadde samtykket til å delta i intervju var til stede ved post-undersøkelsen. Det ble gjennomført to fokusgruppeintervjuer på 15- 20 minutter hver i etterkant av undervisningsopplegget. Under intervjuene satt jeg og de tre elevene rundt et felles bord, med to lydopptakere plassert på hver sin side av bordet. Intervjuene ble tatt opp på lydfil, og jeg tok ikke notater underveis. En av elevene fikk i ansvar å lese opp spørsmålene. Målet var å få elevene til å stå for mesteparten av samtalen, og at de skulle få mulighet til å uttrykke egne tanker og samtidig respondere på det som ble sagt av de andre i gruppa. For å sikre at alle elevene fikk muligheten til å dele sine synspunkter på hver av spørsmålene, henvende jeg meg ved enkelte tilfeller også direkte til enkelte av elevene ved hjelp av navn. Underveis i intervjuet etterstrebet jeg å stille spørsmål som var minst mulig ledende (Postholm & Jacobsen, 2018), og jeg sørget avslutningsvis for å spørre alle informantene om det var noe mer de ønsket å tilføye før vi avsluttet intervjuet, slik jeg finner støtte for i metodelitteraturen (Kvale & Brinkmann, 2015).

Observasjon

Ved å benytte observasjon som metode kan man få beskrivelser av *det som skjer* i klasserommet. Kombinasjonen av spørreskjema, intervju og observasjon kan i følge metodelitteraturen gi mer utfyllende beskrivelser av situasjoner og temaene vi er interessert i å studere (Dalland et al., 2021). Underveis i gjennomføringen av undervisningsopplegget ble det skrevet ustrukturerte observasjonsnotater av meg og faglærer. Etter gjennomføringen utvekslet jeg og lærer erfaringer, før det ble skrevet et fyldigere observasjonsnotat kort tid etter timen. Et sammendrag av noen av de viktigste elementene er inkludert i resultatkapitlet under RQ3. Observasjonene vil ikke bli brukt som del av analysen, men heller som et bidrag inn for å skape et mer helhetlig og nyansert bilde av fenomenet som undersøkes.

4.3 Analyse

I dette delkapitlet følger en beskrivelse av den kvantitative og kvalitative analyseprosessen. De åpne spørsmålene fra spørreskjemaene og transkripsjonene fra fokusgruppeintervjuene ble kodet kvalitativt, og utgjør hoveddelen av analysen. Videre utgjør elevenes graderte påstander ved pre- og postundersøkelsen grunnlaget for den kvantitative analysen. Da jeg i resultatkapitlet presenterer de kvantitative resultatene før de kvalitative, vil jeg følge samme struktur her.

4.3.1 Kvantitativ analyse

Det ble gjennomført to statistiske analyser med utgangspunkt i det kvantitative datamaterialet. Innledningsvis ble de 20 påstandene gruppert med utgangspunkt i rammeverket utviklet for den kvalitative analysen (se «4.3.2 Kvalitativ analyse»), slik det er vist med eksempler i tabell 4. En fullstendig oversikt over påstander og gruppering finnes vedlagt (vedlegg I).

Tabell 4: Eksempel på hvordan de ulike påstandene ble sortert under de forskjellige holdningsdimensjonene utviklet i den kvalitative analysen.

Holdningsdimensjon	Eksempel på påstand
(1) Følelsesmessig innstilling	«Jeg liker programmering»
(2) Oppfatning av egen kompetanse	«Jeg opplever å mestre oppgaver med programmering»
(3) Syn på programmering	«Det er ikke viktig å kunne programmere i biologitimene»

Etter grupperingen ble påstandene videre vurdert og kategorisert som enten *positive* eller *negative*, basert på definisjonene presentert i tabell 5. Tabell 5 viser eksempler på påstander i hver av kategoriene.

Tabell 5: Eksempel på hvordan påstandene i spørreskjemaene ble vurdert som enten positive eller negative.

Type påstand	Definisjon	Påstand
Positiv	Påstander der en høy gjennomsnittsverdi viser til en positiv følelsesmessig innstilling, oppfatning av egen kompetanse som høy eller positivt syn på programmering.	«Jeg liker programmering»
		«Jeg har tro på at jeg kan løse oppgaver med programmering»
		«Det er nyttig å kunne programmere»
Negativ	Påstander der en høy gjennomsnittsverdi viser til en negativ følelsesmessig innstilling, oppfatning av egen kompetanse som lav eller negativt syn på programmering	«Jeg blir nervøs bare av tanken på å skulle programmere»
		«Jeg synes programmering er vanskelig»
		«Det er IKKE viktig å kunne programmere i biologifaget»

Ved en sammenligning av resultatene fra pre- og post-undersøkelsen, vil ønsket effekt av undervisningsopplegget på bakgrunn av definisjonen komme til uttrykk ved en *høyere gjennomsnittsverdi* på de positive påstandene, og en *lavere gjennomsnittsverdi* på de negative påstandene.

Funnene fra den kvantitative analysen presenteres i all hovedsak gjennom *deskriptiv statistikk*, en form for statistisk analyse som beskriver fordelinger ved hjelp av statistiske mål som sentraltendens og spredning (Postholm & Jacobsen, 2018), som gjennomsnitt og standardavvik.

Som et resultat av at elevene i stor grad var «samlet»/enige i sine besvarelser, ble det valgt å *ikke* inkludere et spredningsmål i beskrivelsen av datamaterialet. Dette kom frem av fordelingen av elevenes graderinger av påstandene, og støttes av resultatene fra den kvalitative analysen, som viste forholdsvis liten grad av variasjon i elevenes besvarelser. Det er derfor rimelig å anta at et slikt spredningsmål ville hatt begrenset verdi i tolkningen av resultatene, og fokuset ble derfor på *tendenser* i datamaterialet.

Når svaralternativer i spørreundersøkelsen er «rangerte», slik som i denne studien, er gjennomsnittet et vanlig mål på sentraltendens. Gjennomsnittet sier noe om det typiske svaret i en fordeling, og kan i dette tilfellet si noe om hvilke holdninger og oppfatninger som kjennetegner «flesteparten av elevene» som deltar i undersøkelsen (Postholm & Jacobsen, 2018). Elevenes graderinger av påstandene ble gjort i form av en 5 steps Likert-skala, og ble kodet om til tallverdier mellom 1 – 5, der tallverdien 1 tilsvarer *helt uenig*, og verdien 5 *helt enig*. Verdien 3 tilsvarer et *nøytralt midtpunkt*. Fordelingen av elevenes graderinger vil i resultatkapitlet presenteres med *gjennomsnittsverdi* og *tilhørende verdi* i elevenes spørreskjema. Som eksempel fikk påstanden «Det er nyttig å kunne programmere» en samlet gjennomsnittsverdi på 4,0 som tilsvarer verdien *enig* i spørreskjemaet.

For å vurdere om den observerte forskjellen blant elevenes rapporterte holdninger var signifikant, ble det som del av den kvantitative analysen gjennomført en parvis T-test for å vurdere om forskjellen mellom pre- og post-undersøkelsens gjennomsnitt var signifikant eller ikke (Cohen et al., 2017). Det stilles flere krav til en slik statistisk analyse for å sikre resultatenes troverdighet, og min studie vil ha en begrensning i form av at den er gjennomført på én enkelt klasse, der $n = 21$. P-verdien som fremkommer av en slik test, sier noe om hvor sannsynlig det er at den observerte forskjellen er tilfeldig eller ikke, der nullhypotesen hevdet at det *ikke* er forskjell i gjennomsnittene før og etter gjennomføringen av undervisningsopplegget. I denne studien ble p-verdier under 0,05 satt som grense for signifikans (Cohen et al., 2017).

4. 3. 2 Kvalitativ analyse

I den kvalitative analysen ble besvarelsen fra de åpne spørsmålene på spørreskjemaene og transkripsjonene fra fokusgruppeintervjuene kodet induktivt ved *tematisk analyse*. Analysen er gjennomført etter de seks stegene ved tematisk analyse som beskrevet av Braun og Clarke (2006). Braun og Clarke (2006) argumenterer for at denne formen for analyse er hensiktsmessig der ønsket er å *identifisere mønstre og sentrale «hovedtrekk» ved forholdsvis store datasett*. I dette kapitlet vil jeg beskrive stegene i analyseprosessen, det abduktive rammeverket og temaene som ble utviklet underveis i prosessen.

Steg 1 i den tematiske analysen handler om å gjøre seg kjent med datamaterialet (Braun & Clarke, 2006), en prosess som startet allerede ved gjennomføringen av intervjuene og transkriberingen av disse. For å forenkle kodingsprosessen ble elevenes skriftlige besvarelser importert inn i NVivo, og transkripsjonene ble gjort direkte i samme program. Jeg leste gjennom datamaterialet flere ganger og noterte meg interessante mønstre i en form for «uformell» koding slik at jeg fikk bedre oversikt over datamaterialet.

I **steg 2** fortsatte den mer «formelle» kodingsprosessen, der elevenes utsagn ble organisert i mindre koder som representerte de mest sentrale elementene i datamaterialet. Kodingen var i all hovedsak induktiv, da kodene oppsto underveis på bakgrunn av datamaterialet (Braun & Clarke, 2006). Tabell 6 viser et eksempel på kodingsprosessen. Denne innledende kodingen ble gjennomført i NVivo, som ga meg oversikt over alle koder underveis i prosessen, samt muligheten til å utforske utsagn med felles meningsinnhold. Jeg kodet alt som kunne være interessant med tanke på forskningsspørsmålene, i en form for *åpen koding*. Denne formen for koding kan være hensiktsmessig i studier der det finnes lite tidligere forskning på området, da den åpnet opp for at nye og interessante elementer kunne åpenbare seg underveis i analysen. Eksempelvis kom det av analysen frem hvordan elevenes tidligere erfaringer var svært avgjørende for deres holdninger til det å programmere, et aspekt jeg tidligere ikke hadde lagt stor vekt på i arbeidet med oppgaven. Jeg valgte på bakgrunn av funnene fra den åpne kodingen å inkludere dette som et eget forskningsspørsmål, RQ1.

Tabell 6: Eksempel på kodingsprosessen. Elevutsagnene ble organisert i koder som representerte utsagnets innhold.

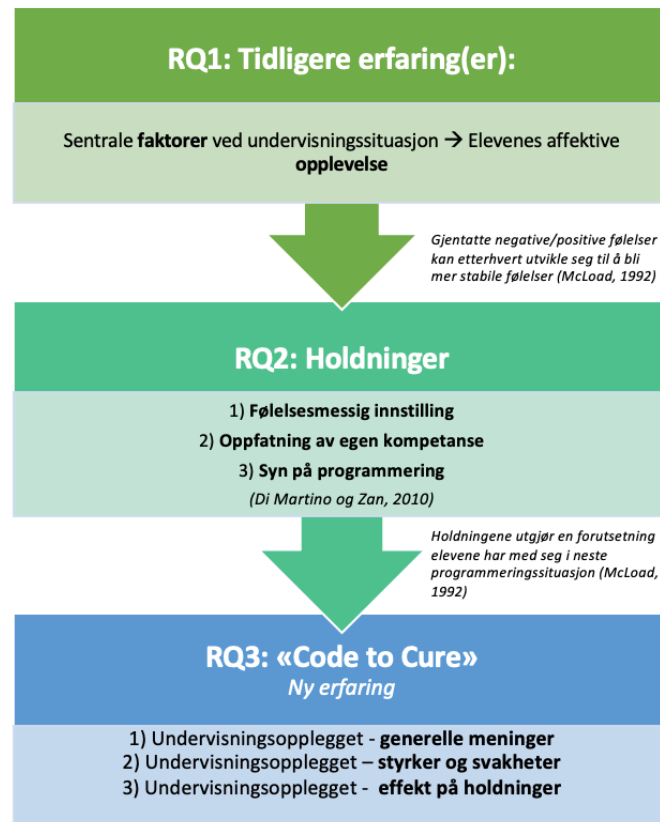
Kode	Elevutsagn
Lærerens programmeringstekniske kompetanse	<i>Det som var mindre bra var at læreren ikke kunne nok om programmering slik at hun/han kunne hjelpe godt nok, eller at man måtte vente lenge på en annen lærer som kunne programmere bedre.</i>
Syn på programmerings nytteverdi i biologi	<i>Jeg vet ikke helt om det (=programmering) bør inn i biologi faget. Sikkert fordi biologi ikke er et regnefag eller noe slik.</i>

Etter den første innledende runden med koding, satt jeg igjen med over 60 innledende koder som samlet ga meg et overblikk over de mest sentrale elementene i datamaterialet. I følge Braun og Clarke (2006) handler **steg 3** om å sortere kodene i mer overordnede *tema*. Videre argumenterer forfatterne for at man underveis i analyseprosessen med fordel kan lage en visuell representasjon av hvilke koder som kan samles i forskjellige tema, og hvordan disse temaene kan sees i forhold til hverandre. Denne prosessen ga meg større oversikt over datamaterialet og forslag til tema og tilhørende koder.

I analyseprosessen ble det ikke gjort et tydelig skille mellom **steg 4 og steg 5**. Det ble gjort en gjennomgang av foreløpige koder, tema og undertema utviklet tidligere i analyseprosessen, som tilsvarer steg 4 (Braun & Clarke, 2006). Parallelt med dette arbeidet ble nye, reviderte temaer definert og navngitt med utgangspunkt i datamaterialet og ulike teoretiske perspektiver, tilsvarende steg 5 (Braun & Clarke, 2006). Denne prosessen ble den mest omfattende og tidkrevende delen av analyseprosessen, der hvert tema ble vurdert hvorvidt temaet virkelig var et tema, om temaet inneholdt nok data, om det var temaer jeg skulle fjerne, slå sammen eller dele opp.

I prosessen valgte jeg å inkludere noen teoretiske perspektiver som kunne bidra inn mot å systematisere mitt foreløpige tematiske kart, og videre utvikle et rammeverk for den reviderte analysen av datamaterialet. McLeod (1992) trekker frem betydningen av tidligere opplevelser på prosessen der holdninger dannes, med særlig fokus på det affektive perspektivet ved opplevelsen. Betydningen av tidligere erfaringer på elevenes holdninger sto sentral i mitt eget datamaterialet, og jeg valgte å fremheve denne sammenhengen med pilene i figur 7.. De tre holdningsdimensjonene beskrevet av Di Martino og Zan (2010), sammenfalt godt med de foreløpige temaene jeg hadde utviklet som beskrivende for elevenes holdninger, og jeg valgte derfor å bruke disse dimensjonene som utgangspunkt når jeg i prosessen reviderte de foreløpige temaene, og konstruerte det abduktive rammeverket presentert i figur 7 og tabellene 7, 8 og 9. Jeg gjorde imidlertid en justering i utviklingen av det abduktive rammeverket, som et resultat av den åpne kodingen. Di Martino og Zan (2010) vurderte «Syn på

matematikk» basert på hvorvidt elevene hadde et instrumentelt eller rasjonelt syn på matematikk. I mitt rammeverk beskriver «Syn på programmering» de forestillingene elevene har til programmering, og er videre definert i tabell 8 og som vedlegg (vedlegg J).



Figur 7: Visuell fremstilling av temaene med utgangspunkt i datamaterialet og teori som dannet utgangspunkt for det abduktive rammeverket presentert i tabellene 7, 8 og 9.

Med utgangspunkt i det foreløpige rammeverket gikk jeg tilbake til det opprinnelige datamaterialet og vurderte hvorvidt temaene fungerte og representerte «det som faktisk skjer» i materialet. Denne prosessen fremheves av Braun og Clarke (2006), og innebærer å gå tilbake til tidligere steg og foreta eventuelle kodinger på nytt med utgangspunkt i rammeverket. Eksempelvis ble koder som «Vi startet for sent» og «Vi kan ikke nok», samlet under koden «Forkunnskaper».

Steg 6, det siste steget i analyseprosessen, innebar å gjennomføre en siste analyse basert på den reviderte oversikten med tema og definisjonene jeg utviklet i tidligere steg. Det blir viktig å presisere at analyseprosessen ikke var lineær slik den er beskrevet her, og at jeg underveis i gjennomføringen beveget meg flere ganger mellom de seks stegene med aktiv lesning av datamaterialet, koding og gjennomgang av tema. For å presentere funnene fra analysearbeidet er rapportering et viktig siste steg i analysen (Braun & Clarke, 2006). Funnene fra denne analysen vil presenteres i oppgavens resultatkapittel, der jeg vil sammenfatte og fremstille funn fra analysen. Braun og Clarke (2006)

fremhever viktigheten av å gjøre dette på en slik måte at man får frem betydningen av hvert tema, gjerne ved hjelp av sitater.

Definisjon av tema med eksempler:

Som en del av steg 5 i analyseprosessen ble de reviderte temaene videre definert, med beskrivelser og eksempler på utsagn. Temaene ble delt inn etter forskningsspørsmål, der temaene 1A «Sentrale faktorer» og 1B «Elevenes affektive opplevelser» beskriver elevenes affektive opplevelser i tidligere møter med programmering, og faktorene som spilte en avgjørende rolle for denne opplevelsen (RQ1). De affektive opplevelsene refererer til elevenes beskrivelser av følelser knyttet til tidligere *spesifikke* undervisningssituasjoner med programmering, i motsetning til tema 2A som representerer holdningskomponenten «Følelsesmessig innstilling». Opplevelsene ble kategorisert som enten positive eller negative, slik det kommer frem av eksempelet i tabell 7.

Tabell 7: Temaene under RQ1 i det abduktive rammeverket med tilhørende definisjoner og eksempler utviklet i analyseprosessen. Eksempler er markert i grått og kursiv.

	Tema	Beskrivelse
RQ1	1A: Sentrale faktorer (tidligere erfaringer)	Lærerens rolle og tilrettelegging <i>Lærerens programmeringstekniske og programmeringsdidaktiske kompetanse, valg av arbeidsmetoder, støttestrukturer, tema og prioriteringer</i> <i>Elevenes forutsetninger, forkunnskaper med programmering</i>
	1B: Elevenes affektive opplevelser (tidligere erfaringer)	Elevenes affektive opplevelser Elevenes («her og nå») følelser ved spesifikke undervisningssituasjoner → mer intense, mindre stabile <i>Følelser som engasjement, mestringsfølelse, frustrasjon og nervøsitet</i> <i>Positive affektive opplevelser: «Det var gøy», «Jeg fikk det til»</i> <i>Negative affektive opplevelser: «Det var kjedelig», «Jeg skjønnte det ikke»</i>

Temaene under RQ2 representerer de tre holdningsdimensjonene, der temaet «Følelsesmessig innstilling» viser til *mer stabile* følelser til programmering, til forskjell fra følelser knyttet til en spesifikk undervisningssituasjon (se tema 1b). Det samme gjeldet temaet «Oppfatning av egen kompetanse», som representerer den mer stabile oppfatningen eleven har om egen kompetanse i programmering, til forskjell fra opplevd mestringsfølelse knyttet til en spesifikk undervisningssituasjon (se tema 1b). Temaet «Syn på programmering» representerer elevenes forestillinger om programmering, og viser til elevenes mer stabile syn på hva programmering er. Utsagn i de tre holdningsdimensjonene ble vurdert som enten positive, negative eller nøytrale (kun for tema 2A), slik det er vist med eksempler i tabell 8.

Tabell 8: Temaene under RQ2 i det abduktive rammeverket med tilhørende definisjoner og eksempler utviklet i analyseprosessen. Eksempler er markert i grått og kursiv.

	Tema	Beskrivelse
RQ2	2A: Følelsesmessig innstilling	Stabile/varige følelser knyttet til programmering (mer enn følelser til spesifikke undervisningssituasjoner) <i>Positiv følelsesmessig innstilling: «Programmering kan være morsomt»</i> <i>Negativ følelsesmessig innstilling: «Jeg liker ikke programmering», «Jeg blir bekymret av å programmere»</i> <i>Nøytral: «Jeg har ikke noe forhold til det»</i>
	2B: Oppfatning av egen kompetanse	Mer stabil/varig oppfatning av egen kompetanse i programmering <i>Lav oppfatning: «Jeg føler ikke jeg mestrer programmering», «Jeg forstår ikke programmering»</i> <i>Høy oppfatning: «Jeg mestrer programmering»</i>
	2C: Syn på programmering	Mer stabil/varig syn på hva programmering er Forestillinger/oppfatninger elevene har om programmering <i>Positivt syn: «Programmering er nyttig i biologi»</i> <i>Negativt syn: «Programmering er ikke nyttig i biologi», «Programmering er for regnefag», «Programmering gjør ting mer komplisert»</i>

Under RQ3, omfatter temaet «Elevenes generelle meninger» elevenes mer «helhetlige» opplevelser og mer generelle meninger om undervisningsopplegget, uten å vise til spesifikke aspekter ved aktiviteten. Styrkene representerer de aspektene elevene mente var bra med undervisningsopplegget, mens svakhetene representerer det elevene opplevde som mindre bra eller forslag til forbedring. Temaet 3C «Effekt på elevenes holdninger» inneholder elevenes refleksjoner på spørsmålet «Har opplegget på noen måte endret deres syn på programmering i biologi – i så fall hvordan?», der elevenes synspunkter er kodet på bakgrunn av de tre holdningsdimensjonene beskrevet i det kvalitative rammeverket.

Tabell 9: Temaene under RQ3 i det abduktive rammeverket med tilhørende definisjoner og eksempler utviklet i analyseprosessen. Eksempler er markert i grått og kursiv.

	Tema	Beskrivelse
RQ3	3A: Undervisningsopplegget - Generelle meninger	Elevenes mer generelle meninger om undervisningsopplegget <i>«Det var spennende», «Jeg likte undervisningsopplegget»</i>
	3B: Undervisningsopplegget - Styrker og svakheter	Konkrete aspekter ved undervisningsopplegget Styrker – hva fungerte bra med opplegget? <i>«Konteksten gjorde meg mer motivert», «Det var nyttig med hint og løsningsforslag»</i> Svakheter – hva kunne vært gjort annerledes? <i>«Opplegget kunne vært mer utfordrende», «Det var en feil i koden»</i>
	3C: Undervisningsopplegget – Effekt på holdninger	Undervisningsoppleggets effekt på elevenes holdninger til programmering Kodet på bakgrunn av de tre holdningsdimensjonene: Følelsesmessig innstilling: <i>«Jeg ble mer motivert for å lære programmering»</i> Oppfatning av egen kompetanse: <i>«Opplegget gjorde at jeg forsto programmering bedre»</i> Syn på programmering: <i>«Opplegget viste meg at programmering kan være nyttig i biologi»</i>

En grundigere forklaring av alle tema og koder ligger vedlagt som vedlegg (vedlegg J).

4. 4 Reliabilitet og validitet

For å kunne avgjøre i hvilken grad vi kan stole på slutninger og funn som legges frem i en studie, er det avgjørende å gjøre rede for studiens *overordnede kvalitet*, ofte omtalt som studiens troverdighet (Frønes & Pettersen, 2021). Troverdigheten kan beskrives ved hjelp av begrepene *reliabilitet* og *validitet*

(Frønes & Pettersen, 2021), og sammen danner de grunnlaget for å si noe om hvorvidt studiens resultater er *gyldige* og *generaliserbare*. Generelt har jeg i oppgaven etterstrebet å inkludere en så grundig beskrivelse av metode og analyse som mulig, slik at leseren forstår hva som er gjort, hvordan og hvorfor. Videre følger en presisering av de valgene som ble tatt for å styrke denne studiens overordnede kvalitet.

Reliabilitet til en studie bestemmes av hvor presise målingene er av fenomenet som undersøkes, og refereres ofte til som studiens *pålitelighet* (Frønes & Pettersen, 2021). Dersom en studie har *høy* reliabilitet, vil en gjentakelse av studien trolig føre til at man kommer frem til samme resultat og konklusjoner. I arbeidet med å styrke reliabiliteten er det i utarbeidingen av spørreskjema og intervjuguide, lagt vekt på å bruke et språk elevene forstår slik at spørsmålene ble tydelige, og det ble etterstrebet å konstruere ikke-ledende, entydige og relevante spørsmål (Postholm & Jacobsen, 2018) for fenomenet som skulle måles. Det ble gjort flere valg for å sikre at elevene fikk mulighet til å uttrykke så presise og pålitelige svar som mulig. Ved de lukkede spørsmålene ble det lagt arbeid i å utvikle så dekkende alternativer som mulig, blant annet ved å inkludere et alternativ for en «nøytral»/ «hverken eller» oppfatning (Brevik & Mathé, 2021; Postholm & Jacobsen, 2018). Det ble etterstrebet å ha en fordeling av både positive og negative påstander, slik at retningen på påstandene varierte (Brevik & Mathé, 2021). De åpne tekstsvarene ga elevene mulighet til å svare mer utfyllende begrunne svarene med sine egne ord (Brevik & Mathé, 2021).

Andre valg ble tatt for å styrke reliabiliteten ved selve datainnsamlingene. Spørreskjemaene var ikke anonyme, og det ble viktig for meg å understreke viktigheten av at elevene turte å svare ærlig på spørsmålene. For å i større grad sikre at elevene turte å dele sine oppriktige følelser og meninger i intervjusituasjonen, ble det tatt et valg om å ha færre elever i fokusgruppene (Tjora, 2021). I intervjusituasjonen stilte jeg oppfølgende spørsmål dersom jeg var i tvil om hva elevene mente, og jeg forsøkte å gjenta elevenes konklusjoner, slik at de fikk mulighet til å kommentere eller korrigere dersom inntrykkene mine var unøyaktige eller det var noe jeg hadde misforstått. Det finnes en sannsynlighet for at elevene under gjennomføringen oppførte seg annerledes fordi det var en fremmed «lærer» til stede (Postholm & Jacobsen, 2018). Dette kunne ha påvirket min helhetlige vurdering av elevenes opplevelser med undervisningsopplegget, og observasjonsnotatene jeg skrev under gjennomføringen. På bakgrunn av dette ble det viktig for meg å klargjøre med faglærer om hen fikk inntrykk av at dette var tilfellet under gjennomføringen, noe det ifølge faglærers inntrykk ikke var grunn til å tro.

For å styrke reliabiliteten i analyseprosessen, har jeg etterstrebet å være så tydelig og transparent som mulig i hvordan jeg har analysert og kodet datamaterialet i studien (Postholm & Jacobsen, 2018).

Kodeskjema er presentert i analysekapitlet, med eksempler på hvilke utsagt som er tillagt ulike tolkninger av meg. Et mer utdypende kodeskjema er også vedlagt som vedlegg (vedlegg J). I resultatkapitlet er det inkludert sitater og eksempler for å illustrere hvordan det kvalitative datamaterialet er tolket. I kvalitativ forskning er det alltid fare for at *forskerbias* påvirker både innsamling og tolkning av datamaterialet – både bevisst og ubevisst (Postholm & Jacobsen, 2018). Derfor er det avgjørende at forskerens subjektivitet legges frem som en del av den konteksten som blir viktig for å forstå studiens funn. Til tross for at man kan argumentere for at all kvalitativ forskning er tolkning, ble det bevisst gjort grep for å gjøre denne påvirkningen så liten som mulig. Jeg merket selv at jeg hadde tro på undervisningsopplegget mitt, og at jeg ønsket at opplegget kunne påvirke elevenes holdninger positivt. Derfor lette jeg bevisst etter indikatorer for ting som ikke fungerte, eller tegn på at elevene hadde mindre positive opplevelser med opplegget. Forhåpentligvis vil en refleksjon og bevisstgjøring omkring egen påvirkning på studiens resultater, samt en synliggjøring av forskningsprosessen ved åpenhet og beskrivelser – føre til at andre kan reflektere over hvordan undersøkelser og jeg som forsker kan ha påvirket resultatet i denne studien.

Studien min bygger på *metodetriangulering*, ved at jeg som forsker gjennom en mixed-methods tilnærming brukte flere metoder for å belyse samme forskningsspørsmål (Brevik & Mathé, 2021). Kombinasjonen kan ifølge litteraturen gi ulike perspektiver som bidrar inn mot å forstå fenomenet bedre (Brevik & Mathé, 2021). Dette ga meg som forsker mulighet til å skape et så helhetlig og nyansert bilde av det jeg ønsket å undersøke. En annen fordel med en slik triangulering, er at kombinasjonen av ulike metoder kan minimere svakhetene ved hver av metodene alene.

Et annet mål på studiens troverdighet, er *validitetsbegrepet* – som sier noe om hvorvidt studien har mestret å måle det den var ment å måle (Frønes & Pettersen, 2021). I forbindelse med validitet, snakker man ofte om en studies *gyldighet*, og for å vurdere dette kan man som forsker stille seg spørsmålet «Hvor sikre er vi på at vi faktisk måler det fenomenet vi sier at vi måler?». Da validiteten avgjøres av relasjonen mellom studiens indikatorer og de begreper som benyttes, blir det *slutningene* man tar på bakgrunn av dataene som avgjør validiteten for undersøkelsen (Frønes & Pettersen, 2021).

For å vurdere om det er samsvar mellom teoretiske begrep og de målingene man gjennomfører, er det vanlig å redegjøre for det som kalles *begrepsvaliditet*. Ved høy begrepsvaliditet sier vi at det er høy korrespondanse mellom begreper og de fenomenene vi faktisk måler (Frønes & Pettersen, 2021). I denne studien handler dette om hvorvidt datamaterialet faktisk reflekterer de holdningene elevene har til å programmere i biologi 2. Flere grep er tatt for å styrke begrepsvaliditeten i studien, og flere av disse handler også her om utarbeidingen av intervjuguide og spørreskjema. Å stille spørsmål som er lett å

tolke for respondentene er viktig for at de skal formulere svar som er på linje med vi ønsker å spørre etter, og det ble etterstrebet å stille tydelige og ikke ledende spørsmål (Postholm & Jacobsen, 2018). Spørsmålene og påstandene i spørreskjemaene ble utviklet med inspirasjon fra allerede utprøvde spørreskjema (Tapia & Marsh, 2002; Yukselturk & Altiok, 2017), som kan være med på å styrke metodens validitet. Det er likevel viktig å være klar over at betydningen på spørsmål og påstander kan forandre noe av sin mening når de blir oversatt fra andre språk og tilpasset denne studiens formål. For å øke sannsynligheten for at målingene i størst mulig grad samsvarer med det jeg ønsket å måle, ble det inkludert flere påstander for å måle samme teoretiske begrep – som for eksempel engasjement eller nytteverdi. Valg av spørsmål, formuleringer og ordlyd ble gjennomgått og diskutert med veiledere før datainnsamlingen fant sted.

Den *ytre validiteten* i en studie handler om hvorvidt resultatene kan overføres – eller generaliseres - til andre utvalg eller situasjoner (Frønes & Pettersen, 2021). Her vil antall og valg av respondenter i undersøkelsen være av betydning. Studien er kun gjennomført i én enkelt biologi 2 klasse, og utvalget varierer mellom 21 – 28 elever avhengig av pre- og post-undersøkelsen. Når utvalget er lite som i denne studien kan dette redusere studiens ytre validitet. På bakgrunn av de undersøkelsene som er gjort, er det ikke urimelig å anta at elevene representerer en generell biologiklasse når det gjelder forkunnskaper og kompetanser i programmering. På denne måten kan funn fra studien forhåpentligvis i større grad bidra med verdifulle perspektiver i liknende kontekster der elevene har få forkunnskaper med programmering fra tidligere.

Statistisk validitet forteller om hvorvidt det er grunn til å anta at funn om sammenhenger er troverdige eller tilfeldige (Cohen et al., 2017). I denne studien var hensikten å undersøke undervisningsoppleggets effekt på elevenes holdninger til programmering. For å måle om en slik forskjell eller effekt er signifikant, ble det valgt å gjennomføre en parvis t-test. P-verdien som fremkommer av en test som denne, sier noe om sannsynligheten for at den observerte forskjellen var tilfeldig eller ikke (Cohen et al., 2017). Resultatene og troverdigheten til undersøkelsen vil avhenge av størrelsen på utvalget, som legger begrensninger på studier som denne.

Studiens troverdighet avhenger av de valgene som er tatt med hensyn til både reliabilitet og validitet. I diskusjonskapitlet «Styrker og svakheter ved studien» vil jeg diskutere hvordan reliabiliteten og validiteten kan ha påvirket denne studiens resultater.

Kapittel 5: Resultater

I dette kapitlet vil jeg presentere resultatene fra analysen av datamaterialet. Kvantitative funn presenteres før kvalitative, og resultatene følger en inndeling tilsvarende oppgavens forskningsspørsmål. Alle sitater i resultatkapitlet vil ha referanser som viser til datakilde og elevnummer. SS1, SS2, SS3 viser til henholdsvis spørreskjema 1, 2 og 3. F1 og F2 viser til fokusgruppeintervju 1 og 2. Eksempelvis vil referansen SS2-4 vise til elev 4 fra spørreskjema 2, og referansen F1-2 vise til elev 2 fra fokusgruppeintervju 1.

5. 1 RQ1 – Tidligere erfaringer og betydningen av disse på elevenes holdninger til å programmere

Her vil jeg presentere funnene som belyser RQ1 - *Hvilke tidligere erfaringer har elevene med programmering i skolen, og hvilken betydning har erfaringene for elevenes holdninger til å programmere i biologi?* Elevenes refleksjoner omkring forholdene «Hva tenker du om at programmering skal inn i biologiundervisningen» og «Hva mener du kjennetegner et godt undervisningsopplegg med programmering i biologi?», synliggjorde betydningen av tidligere erfaringer på elevenes holdninger til å programmere i biologi. På bakgrunn av dette ble det valgt å undersøke tidligere erfaringers betydning for holdninger til å programmere i biologi som et eget forskningsspørsmål – RQ1 – med vekt på (1) hvilke erfaringer elevene har med programmering, (2) kjennetegn ved elevenes tidligere affektive opplevelser, og (3) faktorer ved tidligere erfaringer elevene trekker frem som avgjørende.

Funn fra analysen viser at samtlige av elevene som deltok i undersøkelsen (n=21) hadde tidligere erfaringer med programmering i skolen. Flertallet (n=16) programmerte *ikke* på ungdomsskolen, men av programmeringserfaring på videregående har 48% av elevene programmert litt i ett eller få fag, og de resterende 52% har programmert noen ganger i flere fag. Tabell 10 viser en oversikt over de hyppigst nevnte fagene elevene rapporterer å ha programmert i tidligere.

Tabell 10: Tabell som viser fagene elevene hyppigst har programmert i tidligere. N = 21.

Fag	Nevnt av antall elever
Matematikk	19
Fysikk	9
Kjemi	5
Biologi	2

Av de 19 elevene som besvarte spørsmålet, har samtlige programmeringserfaring fra matematikk, og videre er det realfagene fysikk og kjemi som nevnes flest ganger. Biologi nevnes kun av 2 elever i klassen, til tross for at elevene, ifølge faglærer, gjennomførte en programmeringsaktivitet i biologi 2 noen måneder tidligere. Senere i SS1 blir elevene spesifikt spurt om de har programmert tidligere i biologi 2. I besvarelsene (n = 21), nevner 15 av elevene at de har programmert én gang tidligere i biologi 2, mens 5 elever sier de ikke har programmert i faget tidligere.

I elevenes beskrivelser av tidligere erfaring med programmering i skolen, står to aspekter ved *elevenes affektive opplevelser* sentralt. Elevenes vurdering av tidligere erfaringer ser ut til å være basert på 1) *hvor engasjerte elevene ble i arbeidet med aktiviteten* og 2) *hvorvidt elevene mestret programmeringsaktiviteten eller ikke*. Generelt beskriver elevene lavt engasjement i møte med programmering i undervisningen, og dette kom til uttrykk ved at flere av elevene beskrev tidligere erfaringer lite spennende, lite motiverende og kjedelig. To av elevene åpner likevel opp for at programmering kan være positivt i undervisningen, og at «programmering i timene kan være gøy fordi det gir meg mulighet til å drive med en type problemløsning som man vanligvis ikke får i timen» (SS1-10).

Enda hyppigere nevner elevene grad av mestring som avgjørende element i evalueringen av tidligere programmeringsaktiviteter. Dette kom til uttrykk der elevene beskrev opplevelser knyttet til spesifikke undervisningssituasjoner med «Jeg skjønnte ikke» (SS1-8) «Jeg fikk ikke til» (SS1-1) eller «Jeg ga opp» (SS1-9). I spørreskjemaet der elevene reflekterer over en tidligere erfaring med programmering, skriver flere (n= 13) at programmeringen var vanskelig. Dette ble kodet som uttrykk for lav mestringsfølelse i tilfellene der oppgavens «vanskelighetsgrad» førte til at elevene ikke forsto eller fullførte oppgaven. Fem av elevene skisserer i sine besvarelser en kobling mellom mestring og motivasjon eksplisitt, som i følgende elevsitater:

SS1-1: «Da vi hadde en programmeringsoppgave for noen måneder siden var det veldig få som klarte oppgaven egentlig. Og ting blir bare mindre motiverende om man ikke får det til.»

SS1-2: «Jeg ble ikke super-happy da jeg hørte at vi skulle ha programmering i biologifaget, men jeg skal ikke ta sorgene helt på forskudd. Programmering er mye mer motiverende når man får det til, så det er nok bare litt hjelp man trenger.»

Generelt beskriver elevene lavt engasjement og lav mestringsfølelse i tidligere møter med programmering i undervisningen, og flere viser til en sammenheng mellom grad av mestring og motivasjon til å programmere.

Av faktorer ved tidligere erfaringer som trekkes frem som avgjørende for elevenes opplevelser, er faktorene 1) *lærerens rolle og tilrettelegging*, og 2) *elevenes forutsetninger*, identifisert som særlig betydningsfulle. Over halvparten av elevene trekker frem aspekter ved undervisningen som kan knyttes til lærerens rolle og tilrettelegging, og særlig sentralt står lærerens kompetanse i form av programmeringsteknisk - og programmeringsdidaktisk kompetanse. Flest av elevene har erfaringer med lærere som har lite erfaring med programmering og undervisning med programmering. Følgende sitat trekker frem noen av de utfordringene som kan oppstå når læreren mangler programmeringsteknisk kompetanse:

SS1 - 3: «Det jeg har følt har vært litt lite motiverende er at man på en måte, jeg føler at jeg har vært i så mange situasjoner hvor lærerne starter timen med å si sånn “Jeg kan egentlig ikke programmering selv”, og da er det sånn at man vet at lærerne ikke kan programmering og ikke nødvendigvis heller kan hjelpe deg så mye.»

Enkelte av elevene nevner positive opplevelser med mestring i møte med programmeringsaktiviteter, da ofte som et resultat av at læreren var trygg på selve programmeringen.

Lærers tilrettelegging handler om lærerens valg av rammer og støttestrukturer. Syv av elevene trekker frem undervisningssituasjoner der selve programmeringen innebar å kopiere kode fra lærer eller lærebok, og at dette førte til at elevene «ikke fikk prøvd så mye selv» (SS1- 5) at «det var vanskelig å vite hva man gjorde feil» (SS1-8). Over halvparten av elevene beskriver situasjoner de de ikke forsto oppgaven eller falt av underveis i lærerens gjennomgang fordi programmet ble for langt, gjennomgangen gikk for raskt eller på bakgrunn av at nivået på aktiviteten var lagt for høyt i forhold til hva elevene kunne av programmering. Enkelte av elevene beskriver likevel tilfeller de de på bakgrunn av lærerens tilrettelegging opplevde mestring, og felles for disse opplevelsene var tilfredsstillende støttestrukturer og/eller at læreren prioriterte å bruke god tid på å grundig gjennom programmeringskoden. Av flere elever trekkes det også frem som positivt dersom læreren legger opp til samarbeid, slik at elevene kan diskutere fremgangsmåte og komme med innspill til hverandre.

Den andre faktoren som trekkes frem som avgjørende for tidligere erfaringer, er *elevenes forutsetninger* i form av forkunnskaper. Mangel på forkunnskaper i programmering trekkes frem av samtlige av elevene i pre-undersøkelsen, og flertallet rapporterer å ha for få forkunnskaper til å bruke programmering inn i fagene. Dette illustreres av følgende elevsitat:

SS1 - 3: «Veldig få har hatt programmering fra ungdomsskolen av, men i mange fag forutsetter de allikevel at vi har grunnleggende forståelse og kunnskap om programmering. Når hverken lærere eller elevene er trygge på programmering, blir det vanskelig å utvikle seg, og det blir fort demotiverende.»

5. 2 RQ2 – Elevenes holdninger til å programmere i biologi

For å belyse RQ2 - *Hva kjennetegner elevenes holdninger til å programmere i biologi 2?*, vil jeg med utgangspunkt i det abduktive rammeverket presentert i analysekapitlet, presentere kjennetegn ved elevenes holdninger til å programmere i biologi 2 under én av de tre holdningsdimensjonene – A) følelsesmessig innstilling, B) oppfatning av egen kompetanse og 3) syn på programmering.

Holdningsdimensjon A – Elevenes følelsesmessige innstilling til programmering

Som mål på elevenes følelsesmessige innstilling til programmering, ble det tatt utgangspunkt i respondentenes rangering av påstandene i tabell 11 (se vedlegg I for fullstendig oversikt over påstandene i holdningsdimensjonen).

Tabell 11: Tabell som viser gjennomsnittsverdien til påstandene som måler holdningsdimensjonen «Følelsesmessig innstilling». Skala fra 1 – 5, der 1 = helt uenig og 5 = helt enig. N = 21.

Nr.	Påstand	Gj. snitt
A1	Jeg liker programmering	1,9
A2	Programmering er kjedelig	3,7
A3	Det er spennende å programmere i biologitimen	2,1
A4	Jeg blir mindre motivert av å måtte programmere i biologi	3,9
A5	Jeg blir nervøs bare av tanken på å skulle programmere	3,4

Påstand A1, A2 og A3 beskriver elevenes *engasjement* i møte med programmering i undervisningen. Gjennomsnittsverdiene til de positive påstandene, A1 og A3, ligger begge under det nøytrale midtpunktet, og nærme verdien *uenig* i elevenes spørreskjema. Den negative påstanden, A2, ligger over midtpunktet, nærme verdien som tilsvarer *enig*. Påstand A4 måler elevenes motivasjon for å programmere i biologi, hvor en gjennomsnittsverdi på 3,9 tilsvarer *enig*. Påstand A5 om nervøsitet ligger så vidt over det nøytrale gjennomsnittet. Samlet rapporterer elevene lav motivasjon og lavt engasjement ovenfor det å skulle programmere generelt og i faget.

I datamaterialet der elevene refererte til *mer stabile* følelser ovenfor programmering, ble dette analysert som uttrykk for elevenes følelsesmessige innstilling. Elevenes utsagn ble kodet som enten

positive, negative eller nøytrale, og tabellen nedenfor viser en oversikt over antall ganger innstillingene kom til uttrykk i elevenes besvarelser (tabell 12).

Tabell 12: Kvalitativ koding av elevenes besvarelser der de refererer til holdningsdimensjonen «Følelsesmessig innstilling», N = 21.

Følelsesmessig innstilling	Ganger nevnt
Positiv	2
Negativ	23
Nøytral	3
Totalt	28

Resultater fra analysen viste at flest utsagn ble kodet som negative. Dette kom til uttrykk ved at en stor andel av elevene rapporterte at de ikke liker programmering, at de synes programmering er kjedelig eller ved at de blir nervøse når de skal programmere. En elev beskriver sitt synspunkt til at programmering skal inn i undervisningen på følgende måte:

SS1 – 4: «Jeg synes egentlig ikke noe om det. Det er så mye ork, å gjøre bare forholdsvis enkle ting blir plutselig så komplisert.»

Generelt rapporterer elevene en negativ følelsesmessig innstilling til programmering, og det at programmering skal inn i biologiundervisningen. To av elevene ga uttrykk for en mer positiv innstilling til programmering i undervisningen, men med forbehold om at «det burde begynne med de som har hatt programmering siden ungdomsskolen» (SS1-3) og at elevene ikke burde vurderes på selve programmeringen.

Holdningsdimensjon B – Elvenes oppfatning av egen kompetanse

Tabell 13 viser elevenes graderinger av påstandene som måler holdningsdimensjonen «Oppfatning av egen kompetanse» (se vedlegg I for fullstendig oversikt over påstandene i holdningsdimensjonen). Gjennomsnittsverdien til de *positive påstandene*, B2 og B5, ligger begge *under* det nøytrale midtpunktet og nærme verdien som tilsvarer *uenig* i spørreskjemaet til elevene. Gjennomsnittsverdien til de *negative påstandene*, B1, B3, og B4, ligger alle nærme 4, som tilsvarer verdien *enig*. Et flertall av respondentene vurderer programmering som «vanskelig», og noe de er mindre komfortable med å bruke i undervisningen. Resultatene viser i tillegg at elevene samlet rapporterer lav mestringsfølelse og lav selvtillit når de programmerer, og at et flertall av elevene vurderer sin egen kompetanse i møte med programmering som lav, slik det fremgår i tabell 13.

Tabell 13: Tabell som viser gjennomsnittsverdier til påstandene som måler holdningsdimensjonen «Oppfatning av egen kompetanse». Skala fra 1 – 5, der 1 = helt uenig og 5 = helt enig. N = 21.

Nr.	Påstand	Gj. snitt
B1	Jeg synes programmering er vanskelig	3,9
B2	Jeg opplever at jeg mestrer oppgaver med programmering	1,8
B3	Når jeg programmerer har jeg lite selvtillit	4,1
B4	Når jeg får en oppgave i programmering vet jeg at jeg ikke kommer til å klare den	3,8
B5	Jeg er komfortabel med å bruke programmering i biologiundervisningen	1,7

Der elevene i sine besvarelser viste til et mer *stabilt* syn på egen kompetanse i møte med programmeringsaktiviteter, ble dette tolket som uttrykk for holdningsdimensjonen «Oppfatning av egen kompetanse». Resultatene fra analysen viser at en stor andel av elevene opplever egen kompetanse i møte med programmering som *lav*, og et flertall av elevene refererer i et eller flere av sine besvarelser til programmering som noe *vanskelig* og *noe de ikke får til*. Denne oppfatningen er avgjørende i flere av elevenes begrunnelser for hvorfor de ikke liker programmering, og er negative til at programmering skal inn i biologiundervisningen. Oppfatningen av å ikke mestre oppgaver med programmering er gjennomgående blant elevenes besvarelser, og underbygges av følgende elevsitater:

SS1 - 2: «Fordi ... jeg synes ikke jeg mestrer programmering i det hele tatt, og jeg synes det er veldig lite motiverende. Det gjør at hele, for eksempel i fjor da i matte, ble mye mindre motiverende. Fordi jeg visste at programmering var en stor del av matte og jeg mestrer det ikke på en måte.»

F1 - 2: «Programmering er så nytt, og de nye læreplanene regner jo egentlig med at vi kan mer enn vi egentlig kan. Så da føles det litt sånn lite motiverende ut. Fordi man føler seg egentlig litt dum, og det er jo ikke bra. Det er jo ingen som liker å føle seg dumme.»

Sitatene ovenfor er eksempler på tilfeller der elevene beskriver et mer stabilt syn på egen kompetanse i møte med programmering som lav, i motsetning til å beskrive manglende mestring ved spesifikke undervisningssituasjoner.

Holdningsdimensjon C – Elevenes syn på programmering

Tabell 14 viser elevenes graderinger av et utvalg av påstandene som måler holdningsdimensjonen «Syn på programmering». Påstandene om verdi/nytteverdi ligger alle *over* det nøytrale midtpunktet, med unntak av C4 «Jeg kan tenkte meg mange tilfeller der det kan være nyttig å programmere i biologi». Påstandene rettet spesifikt inn mot biologifaget, C3 og C4, viser noe lavere gjennomsnittsverdier på 3,4

og 2,7, sammenlignet med påstandene som ser på programmering generelt, C1 og C2. Resultatene viser at elevene samlet vurderer kompetanser innen programmering som nyttig, samtidig som de forstår hvorfor man burde lære programmering i skolen. Likevel kan det se ut til at elevene vurderer programmering som mindre viktig i faget biologi 2, sammenlignet med generelt/i andre fag.

Tabell 14: Tabell som viser gjennomsnittsverdier til påstandene som måler holdningsdimensjonen «Syn på programmering». Skala fra 1 – 5, der 1 = helt uenig og 5 = helt enig. N = 21.

Nr.	Påstand	Gj.snitt
C1	Det er nyttig å kunne programmere	4,0
C2	Jeg forstår hvorfor vi burde lære programmering på skolen	3,7
C3	Jeg tror programmering er nyttig hvis man vil studere eller jobbe innenfor biologi	3,4
C4	Jeg kan tenke meg mange tilfeller der det kan være nyttig å programmere i biologi	2,7

Resultatene fra den kvalitative analysen viser at elevenes syn på programmering preges av ulike forestillinger om hva programmering er og i hvilke situasjoner programmering som verktøy har en verdi. De mest gjennomgående forestillingene omhandler programmeringens nytteverdi, generelt og i faget biologi. Flere av elevene (n = 6) nevner programmeringens nytteverdi eksplisitt i sine beskrivelser, og anerkjenner programmeringens nytteverdi, til tross for at flere av de «ikke liker programmering»:

SS1 - 3: «Jeg skjønner at det kan være nyttig, og at mye av fremtiden vi basere seg på programmering, men jeg liker det ikke.»

Likevel ser elevene ut til å være uenige i hvorvidt programmering er nyttig i fag som biologi 2. Elevene som opplever programmering som et nyttig verktøy *generelt*, men betviler programmeringens nytteverdi spesifikt inn i biologifaget, begrunner dette ofte med forestillingen om at programmering er mest nyttig i såkalte «regnefag», som matematikk og fysikk:

SS1 – 5: «Jeg vet ikke helt om det bør inn i biologi faget. Nok fordi biologi ikke er regnefag eller noe slik. Jeg synes derfor det ikke er veldig relevant i biologifaget. Det er nok noen regning, men programmering tilhører mest matte tenker jeg.»

Dette introduserer en annen sentral forestilling elevene har til programmering, nemlig forestillingen om at *programmering er for regning*. Når elevene i SS1 skal nevne tilfeller der det kan være nyttig å programmere i biologi, viser analysen at elevene generelt har få eksempler på bruksområder for programmering i faget. Av eksemplene som presenteres, er samtlige knyttet opp mot situasjoner der man studerer populasjonsvekst, mer generell statistikk (fremstille tabeller, modeller og grafer) og

sannsynlighet. Til tross for denne forestillingen, vurderer flere elever programmering som nyttig, også i biologi. En elev beskriver programmeringens nytteverdi i faget på denne måten:

SS1 - 6: «Jeg tenker at det er flere bruksområder i biolog-yrket, men vet ikke om man burde fokusere så mye på det i selve faget. Flere som tar biologi skal ikke bli biologer eller relaterte yrker, men dette betyr ikke at programmering skal være helt ekskludert fra faget.»

En annen forestilling som ble synlig gjennom analysen handler om elevenes syn på programmering som noe «unødvendig» og noe som «gjør ting mer komplisert», to forestillinger som støttes av følgende sitater:

SS1 - 7: «Jeg har opplevd at på det nivået vi er nå, med programmering, er det ikke særlig mange oppgaver der det lønner seg å bruke programmering. Programmer som GeoGebra kan løse alle oppgaver på vårt nivå - om man er lite motivert i utgangspunktet, sliter man kanskje med å se fordelene med f. eks "Python > Geogebra".»

SS1 - 4: «Programmering er så mye ork, og å gjøre bare forholdsvis enkle ting blir plutselig så komplisert.»

I det første sitatet beskriver eleven tilfeller der andre verktøy som ikke er programmering hadde fungert like godt eller bedre. Eleven SS1-4 beskriver hvordan bruk av programmering i undervisningen kan gjøre «forholdsvis enkle ting» mer kompliserte, og at dette påvirker elevenes syn på verktøyet. Flere av elevene (n = 6) beskriver følelsen av å bli overveldet når de i programmeringsaktiviteter må forstå og forholde seg til både det faglige og det programmeringstekniske, og hvordan programmering ofte legges til de fagene som allerede er «kompliserte nok fra før», slik som beskrevet av én av elevene under fokusrupointervjuet:

F1-1: «Og i tillegg så føler jeg programmering er i de veldig "vanskelige" fagene, som er så teoretiske og matematikk vanskelig fra før, som R-matte og fysikk og sånn. Så da er det nok å nesten bare lære seg det faget, så får man programmering på toppen av det hele. Så da blir det demotiverende, når man skal lære SÅ mye nytt, da.»

Eleven beskriver programmering som noe ekstra som kommer «på toppen» av det faglige, og da ofte i de fagene eleven opplever som utfordrende nok fra før av. Denne oppfatningen av programmering «som noe ekstra», underbygges av følgende sitat fra en annen elev:

SS1 - 7: «Når vi blir introdusert for programmering så sent i skolegangen, gir dette følelsen av «Er det enda en ting vi må lære?! Blir vi aldri ferdige?!»

Samlet viser resultatene hvordan elevenes syn på programmering i all hovedsak kan beskrives med forestillingene «Programmering som nyttig», «Programmering er for regning», «Programmering er unødvendig» og «Programmering blir noe ekstra i fagene». Andre syn på programmering som sto sentralt i elevenes besvarelser var «Programmering er kjedelig» og «Programmering er vanskelig». Disse forestillingene har jeg valgt å presentere som en del av resultater under de to tidlige holdningsdimensjonene, henholdsvis «Følelsesmessig innstilling» og «Oppfatning av egen kompetanse».

5. 3 RQ3 – elevenes opplevelser med undervisningsopplegget «Code to Cure»

For å belyse oppgavens siste forskningsspørsmål RQ3 - *Hvordan påvirkes elevenes holdninger til å programmere i biologi 2 av det kontekstbaserte undervisningsopplegget «Code to Cure»?», vil jeg innledningsvis presentere et sammendrag av egne og faglærers observasjoner fra gjennomføringen. Videre følger en presentasjon av elevenes opplevelser med undervisningsopplegget «Code to Cure», med utgangspunkt i følgende inndeling 1) Elevenes generelle meninger om undervisningsopplegget «Code to Cure», 2) Styrker og svakheter ved undervisningsopplegget og 3) Undervisningsoppleggets effekt på elevenes holdninger til å programmere i biologi 2.*

5.3.1 En sammenfatning av egne og faglærers observasjoner fra gjennomføringen

Følgende *observasjonsnotat* er basert på egne og faglærers observasjoner under gjennomføringen, og er inkludert i resultatene for å gi et mer helhetlig bilde av elevenes opplevelser med undervisningsopplegget. Generelt opplevde både jeg og faglærer elevene som jevnt over engasjerte og motiverte i arbeidet med undervisningsopplegget. For å være tilgjengelig for eventuelle spørsmål og å observere elevenes arbeidsprosesser, beveget jeg meg mellom de ulike gruppene underveis i økta. Det var generelt få spørsmål til meg og faglærer, og de fleste spørsmålene som kom var i forbindelse med en teknisk feil i det ene programmet, som imidlertid ble løst raskt. Enkelte av gruppene valgte å gjennomføre opplegget på én felles skjerm, andre valgte å benytte seg av hver sin PC og jobbe med opplegget parallelt.

Elevenes diskuterte aktivt og dialogene innad i gruppene var i all hovedsak faglige. Både innad og på tvers av gruppene, diskuterte elevene på eget initiativ både tid på eksperimentet og symptomer, og sammenlignet sine funn med hverandre. Mange grupper brukte godt tid på aktiviteten «Finn den skeptiske legen», der én av elevene ble trukket tilfeldig for å gjennomføre et eksperiment. Flere av

elevene ga uttrykk for å være overrasket når de innså hvor lang tid det ville ta å lese gjennom hele gensekvensen manuelt i eksperimentet.

Alle gruppene ble ferdig med hoveddelen av opplegget der de skulle diagnostisere og behandle pasient, og omtrent halvparten kom i gang med ekstraoppgaven. De resterende gruppene så ut til å si deg fornøyd med å være ferdig med hovedoppgaven, og begynte etter hvert å ha samtaler utenom det faglige. I løpet av gjennomføringen fortalte to elever at de likte opplegget fordi de fikk mye av koden på forhånd, og at oppgaven heller handlet om å legge til og endre enkelte elementer. En av de kom med følgende refleksjon:

Elev under gjennomføringen: «Jeg føler det er mer ... motiverende på den måten. For ofte vet vi ikke helt hvordan vi skal komme i gang, og da blir jeg ofte bare sittende uten å gjøre noe.. Så blir man frustrert av å ikke få det til».

Hos to av gruppene så ansvaret med programmeringen ut til å være satt til én elev, noe som førte til at de resterende elevene på gruppa opplevdes som mer passive og mindre engasjerte i situasjonene der de måtte programmere. Hovedoppfatningen til meg og faglærer var likevel at elevene jobbet godt sammen innad i gruppene, også i delene der de måtte programmere.

5.3.2 Elevenes generelle meninger om undervisningsopplegget «Code to Cure»

Tabell 5 viser påstandene elevene graderte i post-undersøkelsen, hvor flere av gjennomsnittsverdiene gir et utgangspunkt for å beskrive elevenes *generelle meninger om undervisningsopplegget*. Påstand U1 er gradert til gjennomsnittsverdien 4,5 som tilsvarer et sted mellom *enig* og *helt enig* på elevenes spørreskjema. Gjennomsnittsverdien til påstand U2 tilsvarer verdien for *enig*, som underbygges av gjennomsnittsverdien til den reverserte påstanden om motivasjon, U3. Hvorvidt opplegget førte til at elevene lærte noe, kommer frem av elevenes graderinger av påstand U4. Et gjennomsnitt på 3,8 ligger nær *enig*, som viser til at opplegget kan ha bidratt til at flere av elevene lærte noe i arbeidet med aktiviteten.

Tabell 15: Tabell som viser gjennomsnittsverdien til påstandene om elevenes generelle meninger om undervisningsopplegget «Code to Cure». Skala fra 1 – 5, der 1 = helt uenig og 5 = helt enig. N = 28.

Nr	Påstand	Gj. snitt
U1	Jeg likte undervisningsopplegget	4,5
U2	Opplegget gjorde meg mer motivert for å programmere	3,9

U3	Jeg mistet motivasjonen når jeg jobbet med programmeringsoppgavene	1,9
U4	Programmeringen førte til at jeg lærte noe	3,8

Resultatene fra den kvalitative analysen åpnet opp for et mer nyansert bilde av elevenes meninger, hvor et flertall av elevene beskriver positive opplevelser med undervisningsopplegget. Generelt rapporterte elevene at de likte timen, og flere av elevene (n = 17) beskrev opplegget og konteksten som spennende og motiverende. Følgende sitat er fra en av elevenes refleksjoner under fokusgruppeintervjuet:

SS2 – 1: «Jeg syntes oppgaven ble mye mer spennende når det var satt i en realistisk kontekst. Samtidig som meningen er å lære programmering ble det mer engasjerende når det var delt inn i oversiktlige oppgaver, og når det var et slags "mål" med programmeringen.»

Til tross for at elevene hadde flere innspill på hva som kunne vært forbedret med opplegget, fant jeg ingen utsagn blant elevenes besvarelser som ga uttrykk for *generelle* negative meninger om undervisningsopplegget. Elevenes forslag til forbedringer presenteres i neste avsnitt under svakheter ved undervisningsopplegget.

5.3.3 Styrker og svakheter ved undervisningsopplegget

En sammenfatning av de mest sentrale faktorene som ble avgjørende for elevenes opplevelser med undervisningsopplegget, vil her presenteres i form av *styrker og svakheter ved opplegget*. Graderingen av påstandene i tabell 16 presenterer elevenes samlede vurdering av flere konkrete aspekter ved undervisningsopplegget som nivå, hvorvidt opplegget var enkelt å forstå, arbeidsmetode og støttestrukturer. Basert på gjennomsnittsverdiene til påstand U5 og U6 ser det ikke ut til at elevene opplevde nivået – hverken på det biologifaglige eller på selve programmeringen – som for høyt. Gjennomsnittsverdien til påstand U7 ligger nær det nøytrale midtpunktet, og fordelingen av elevenes svar viser at flest elever graderte påstanden til verdien «nøytral». Gjennomsnittsverdien til påstandene U8 og U9, tilsvarer verdien *enig* i hvorvidt opplegget var enkelt å forstå, og hvorvidt støttestrukturene, som hint og løsningsforslag, var nyttige. Det ser ut til at elevene generelt opplevde strukturen på opplegget som god, og at flertallet var fornøyd med å jobbe i grupper og uten for mye felles gjennomgang fra lærer, påstand U10 og U11.

Tabell 16: Tabell som viser gjennomsnittsverdien til påstandene om styrker og svakheter ved undervisningsopplegget «Code to Cure». Skala fra 1 – 5, der 1 = helt uenig og 5 = helt enig. N = 28.

Nr	Påstand	Gj. snitt
----	---------	-----------

U5	Nivået på det faglige (biologi-delen) var på et for høyt nivå	1,8
U6	Nivået på programmeringen var på et for høyt nivå	1,6
U7	Opplegget var ikke utfordrende nok	2,6
U8	Det var lett å forstå hva vi skulle gjøre underveis i opplegget	4,3
U9	Det var nyttig med hint og løsningsforslag på programmeringsoppgavene	3,7
U10	Jeg ville likt opplegget bedre dersom vi jobbet individuelt	2,0
U11	Jeg ville likt opplegget dersom vi gjorde mer felles (med lærer)	2,2

Flere av elevene nevner *samarbeid, struktur på opplegg og valg av støttestrukturer* som komponenter ved opplegget som fungerte bra. Faktorene som helt klart nevnes flest ganger er kodet under *kontekst og rammer og støttestrukturer*. Resultatene viser at samtlige av de 28 elevene vurderte *konteksten* for programmeringsaktiviteten som positiv. Videre beskriver halvparten av elevene (n = 14) hvordan historien bak bidro til økt engasjement og motivasjon for programmering. Følgende elevsitat understreker dette:

SS2 – 2: «Jeg syntes at opplegget var veldig bra og spennende. Vanligvis så er jeg ikke så glad i programmering, for eksempel i matte, men når det ble satt sammen og i kontekst med noe jeg interesserer meg for ble det spennende. Da syntes jeg heller ikke at det var like vanskelig som det pleier å være.»

Flere av elevene beskriver hvordan konteksten bidro med å få frem verktøyets nytteverdi i faget, ved at elevene ble introdusert for en helt spesifikk situasjon der programmering er verdifullt i dagens biologi, slik som i følgende sitat:

SS2 – 3: «Jeg synes opplegget var veldig bra. Når vi har programmert i matte eller fysikk for eksempel har vi gjort oppgaver der det er litt vanskelig å skjønne hensikten med å programmere. Ofte kan vi i de tilfellene gjøre det på en enklere måte i CAS eller forhånd. I dette opplegget derimot ble jeg overbevist om at programmering var et lettere alternativ. I tillegg likte jeg at det ble satt inn i en praktisk og virkelighetsnær situasjon.»

Analysen viser at elevene refererer til koden *rammer og støttestrukturer* som en positiv faktor hele 50 ganger i deres evalueringer av opplegget. Elevene nevner egenskapen med at opplegget var *enkelt å forstå og følge*, som særlig avgjørende for deres positive opplevelser med aktiviteten. Samtidig rapporterer et flertall av elevene at de likte strukturen med nettside og læringssti-heftet, og at de satte

pris på støttestrukturene som hint, eksempler underveis og løsningsforslag til hver oppgave. Dette underbygges av følgende elevsitater:

SS2 – 3: «Jeg likte at vi fikk gode instruksjoner både i heftet og på nettsiden om hva vi skulle gjøre. I tillegg likte jeg at vi fikk kodene i de første programmeringsoppgavene, mens vi i den siste måtte bruke de andre kodene og det vi hadde lært på egenhånd.»

SS2 – 4: «Jeg synes at opplegget var veldig godt strukturert og forståelig. Man skjønte hele tiden hva man skulle gjøre, og det var bra at man fikk hint og fasit.»

Flere av elevene trekker frem faktorene *variasjon*, *samarbeid* og *nivå* som styrker ved undervisningsopplegget. Åtte av elevene beskriver at de likte hvordan opplegget ikke *kun* besto av programmeringsoppgaver, men at det var en variasjon av ulike oppgaver – som å lete etter symptomer og sammenligne sykdomsbilde ved flere aktuelle sykdommer. En elev skriver at «Det var mer motiverende enn programmering vanligvis er, fordi vi hadde et mål vi skulle nå og ikke hele oppgaven var programmering, så vi fikk et avbrekk her og der» (SS2-6). Dette synspunktet støttes av følgende sitat fra en annen elev:

F2 - 2: «Altså for min del, så tenker jeg det at jeg.. gir ofte opp bare jeg hører.. med en gang jeg hører ordet programmering. Men når det var et opplegg hvor ... man på en måte, ikke begynte rett på med programmering, men man fikk litt sånn, start-oppgaver og sånn først. Før man begynte med programmering. Det synes jeg var litt fint, for da begynte jeg å bli investert i å fullføre oppgaven.»

Flere av elevene (n = 10) trakk frem *samarbeid* som en styrke ved undervisningsopplegget, og begrunner dette med at programmeringsaktivitetene «ofte blir mer overkommelige når man kan snakke sammen og lære av hverandre» (SS2-7). Med unntak av én elev som opplevde siste oppgave av ekstraoppgaven som for utfordrende, var det ingen tilbakemeldinger fra elevene som handlet om at opplegget var på et for høyt programmeringsteknisk eller faglig nivå. Flere av elevene (n = 6) beskriver eksplisitt hvordan nivået på opplegget var på et passende nivå, ved at programmeringen ble grundig forklart og ikke krevde et for høyt nivå av forkunnskaper fra elevene:

SS2 – 5: «Jeg synes det var en fin måte å jobbe på. Det var ikke for mye "hard" programmering, men det var eksempler og forklaringer. Det var fin innledning, også ble det vanskeligere utover i heftet.»

De presenterte faktorene er valgt på bakgrunn av hvor ofte de er nevnt i elevenes besvarelser. En oversikt over samtlige av styrkene og svakhetene elevene rapporterte, er samlet i tabell 17.

Tabell 17: Styrker og svakheter ved undervisningsopplegget «Code to Cure», som rapportert av elevene. N = 28.

Styrker ved undervisningsopplegget	Aktivt Allerede kjent tema Biologisk problemstilling Kontekst Passende nivå Nok tid Fremhevet nytteverdi Personlig Rammer og støttestrukturer Samarbeid Variasjon
Svakheter ved undervisningsopplegget	Kunne vært mer utfordrende Konkret om nettsiden Feil i kode For mye forklaring For lite forklaring For mange på en gruppe Savnet felles gjennomgang

Svakheterne ved undervisningsopplegget kom til uttrykk gjennom elevenes besvarelser på spørsmålet «Hva fungerte mindre bra med undervisningsopplegget? Hva kunne vært gjort annerledes?». Tabell 18 viser en oversikt over aspektene ved undervisningsopplegget elevene beskrev som mindre bra, eller som forslag til forbedring. Seks av elevene nevner ved post-undersøkelsen at programmeringen i opplegget med fordel kunne vært mer utfordrende, og utgjør med dette den faktoren som er nevnt flest ganger i elevenes besvarelser. Følgende utsagn er hentet fra elevenes besvarelser:

F2 - 2: «Det var kanskje litt lite utfordrende, faktisk (ler). Selv om jeg ikke er noe super-programmerer, så var det nok litt – jeg synes det var litt mye hint, på en måte(...). Så ... det virka kanskje mer som en sånn "oppstart" for programmering, mer enn at du anvender programmeringskunnskap du har fra før. Så kanskje noen av disse "Du skal gjøre dette og dette", skulle vært litt mer gjemt?»

Eleven beskriver her hvordan de grundige forklaringene underveis med fordel kunne vært unngått, slik at det i større grad hadde åpnet opp for at elevene fikk tenke mer på egenhånd. Flere av elevene (n = 7) hadde konkrete tilbakemeldinger til nettsiden, som at sykdommene var litt vanskelige å lese og «Nettsiden hoppet slik at man kom rett inn på programmeringsoppgavene og ikke fikk lest det som sto før programmeringen ... Men man måtte bla opp selv» (F2-3). Andre svakheter ved opplegget og forslag fra enkeltelever er presentert i tabell 17.

5.3.4 Undervisningsoppleggets effekt på elevenes holdninger til programmering i biologi 2

For å undersøke om undervisningsopplegget «Code to Cure» hadde en effekt på elevenes holdninger til å programmere i biologi 2, er det tatt utgangspunkt i følgende: 1) elevenes graderinger av påstander om undervisningsopplegget, 2) en sammenligning av elevenes graderinger av påstandene før og etter gjennomføringen og 3) elevenes refleksjoner i fokusgruppeintervjuene.

Tabell 19 viser elevenes graderinger av påstandene om undervisningsopplegget «Code to Cure», der fire av påstandene måler oppleggets effekt på motivasjon og syn på programmering som et nyttig verktøy. Påstand U2 og U4 ble begge gradert til gjennomsnittsverdier på 3,8, som ligger nært verdien for *enig* i elevenes spørreskjema. Påstand U12 har en gjennomsnittsverdi på 4,2, som ligger over verdien for *enig*. Påstand U8 har en gjennomsnittsverdi på 4,6 som tilsvarer et sted mellom *enig* og *helt enig* i elevenes spørreskjema.

Tabell 18: Tabell som viser gjennomsnittsverdien til påstandene om effekten av undervisningsopplegget «Code to Cure». Skala fra 1 – 5, der 1 = helt uenig og 5 = helt enig. N = 28.

Nr	Påstand	Gj. snitt
U2	Opplegget gjorde meg mer motivert for å programmere	3,9
U4	Programmeringen førte til at jeg lærte noe	3,8
U12	Jeg forstår hvorfor programmering var en del av undervisningsopplegget	4,2
U13	Jeg forstår hvorfor programmering er viktig for å analysere store mengder data (for eksempel DNA-sekvenser)	4,6

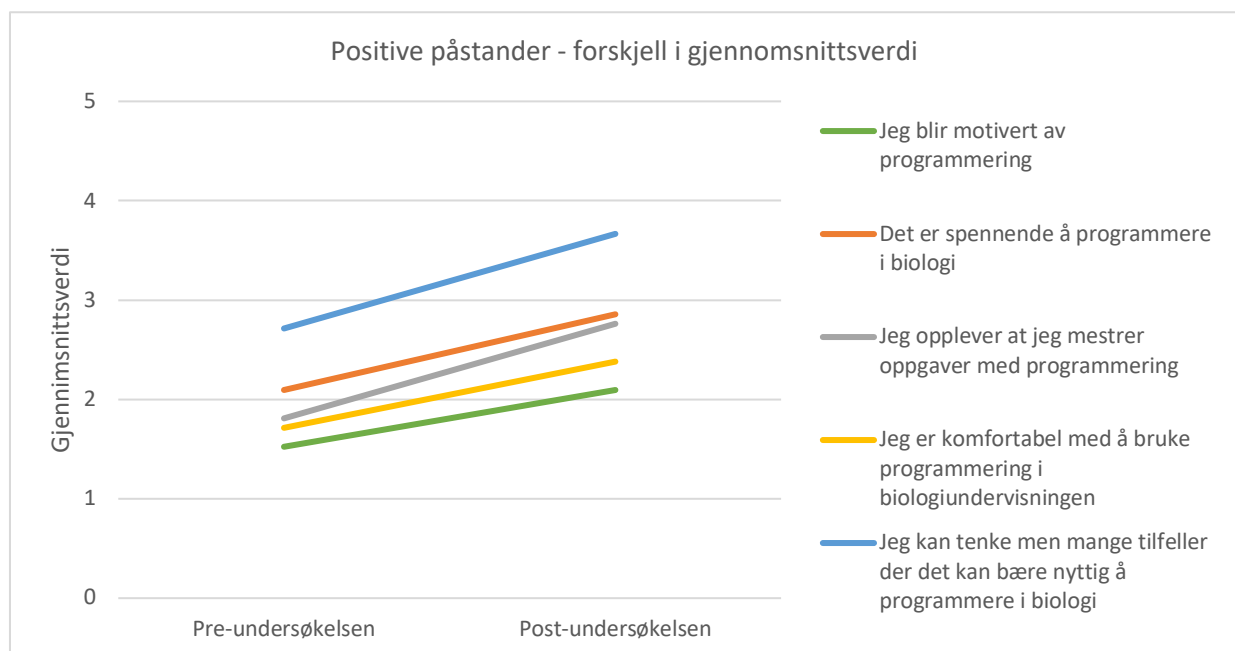
Resultatene fra sammenligningen at elevenes graderinger av påstandene før og etter gjennomføringen, viser at alle de *positive* påstandene har en høyere gjennomsnittsverdi, og alle de *negative* påstandene har en lavere gjennomsnittsverdi ved post-undersøkelsen, sammenlignet med tidligere. Funnet peker mot det som i analysekapitlet ble definert som «ønsket effekt» på elevenes holdninger, og gjelder for påstandene gradert umiddelbart etter gjennomføringen ved SS2, og påstandene gradert tre uker etter gjennomføringen av undervisningsopplegget, SS3.

Resultatene fra den *forenklede* parvise t-testen fant p-verdier < 0,05 på 8 av de totalt 20 påstandene. Tabell 19 viser et utvalg av de totalt 20 påstandene hvor p-verdi ble beregnet til under 0,05. En oversikt over alle påstander og tilhørende p-verdi finnes vedlagt (vedlegg I).

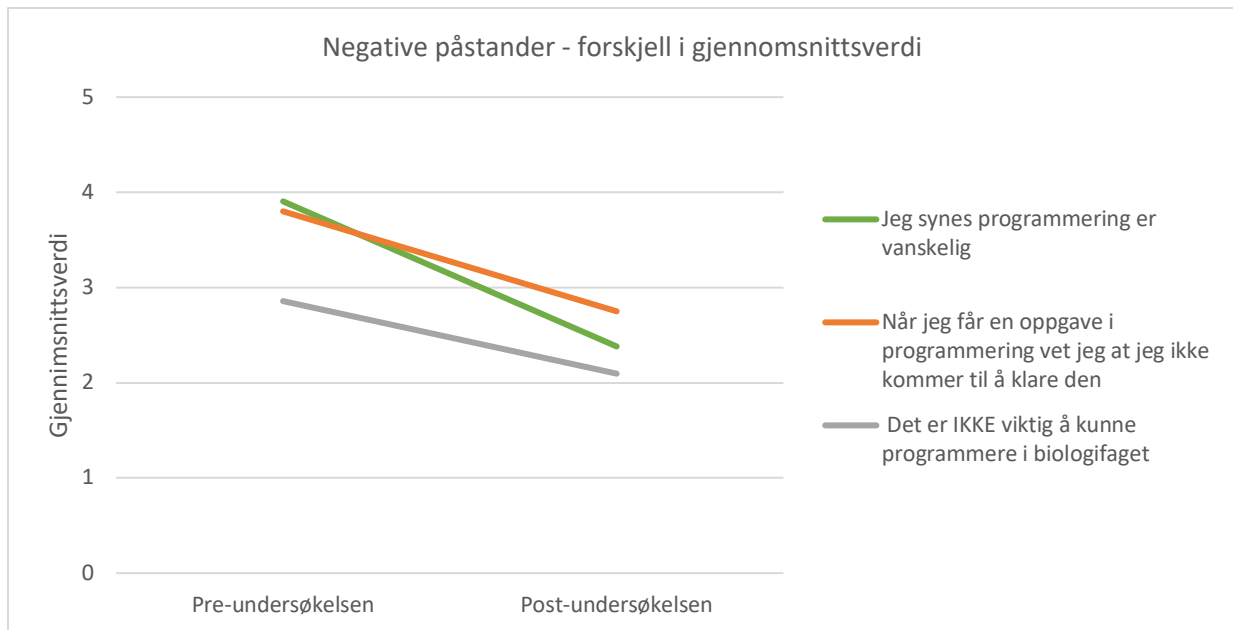
Tabell 19: Oversikt over påstandene som viser størst sannsynlighet for signifikant forskjell på gruppegjennomsnitt ved pre-test og post-test (p-verdi < 0,05). Post-test (SS2) viser resultatene fra spørreskjemaet elevene besvarte umiddelbart etter gjennomføringen.

Holdningsdimensjon	Påstand	Pre-test (SS1)	Post-test (SS2)	Post-test (SS3)	p-verdi
(1) Følelsesmessig innstilling	Jeg liker programmering	1,5	2,1		0,0422
	Det er spennende å programmere i biologi	2,1		2,9	0,0077
(2) Oppfatning av egen kompetanse	Jeg synes programmering er vanskelig	3,9	2,4		0,0005
	Jeg opplever at jeg mestrer oppgaver der jeg skal programmere	1,8	2,8		0,0035
	Når jeg får en oppgave der jeg skal programmere vet jeg at jeg ikke kommer til å klare den	3,8	2,75		0,0270
	Jeg er komfortabel med å programmere i biologiundervisningen	1,7		2,4	0,0155
(3) Syn på programmering	Det er IKKE viktig å kunne programmere i biologifaget	2,9	2,1		0,0032
	Jeg kan tenke meg mange tilfeller der det kan være nyttig å programmere i biologi	2,7		3,7	0,0043

Forskjellen i gjennomsnittsverdier ved de åtte påstandene er presentert nedenfor i figur 8 og figur 9. Figur 9 viser en oversikt over de positive påstandene, der økt gjennomsnitt viser til positiv effekt på holdninger. Figur 9 viser de tre negative påstandene, der et lavere gjennomsnitt viser til en positiv effekt på holdningene.



Figur 8: Forskjell i gjennomsnittsverdi ved pre- og postundersøkelsen (positive påstander). Høyere gjennomsnitt indikerer en positiv effekt på elevenes holdninger. N = 21.



Figur 9: Forskjell i gjennomsnittsverdi ved pre- og postundersøkelsen (negative påstander). Lavere gjennomsnitt indikerer en positiv effekt på elevenes holdninger. N = 21.

De kvalitative resultatene er i hovedsak basert på analysen av de transkriberte fokusgruppeintervjuene, der elevene diskuterte hvorvidt opplegget hadde en effekt på deres holdninger til å programmere i biologi. Analysen viser at samtlige av elevene rapporterer at opplegget hadde en effekt på deres holdninger til å programmere i faget, og en sammenfatning av utsagnene er presentert i tabell 20 og gruppert på bakgrunn av de tre holdningsdimensjonene.

Tabell 20: Noen av effektene på holdninger elevene rapporterte etter gjennomføringen av «Code to Cure», kategorisert under de tre holdningsdimensjonene.

Effekt på (1) følelsesmessig innstilling	Mer motivert for å lære programmering Viste at programmeringen kan være morsomt
Effekt på (2) oppfatning av egen kompetanse	Forsto programmering bedre Fikk bedre oversikt over programmering
Effekt på (3) syn på programmering	Programmering kan være nyttig i biologi Programmering kan være noe mer enn bare «noe ekstra» → Kan bidra til å forstå faget bedre

En elev skriver i sin evaluering av undervisningsopplegget at det gjorde hen mer motivert til å lære programmering når det var del av et slikt undervisningsopplegg. En annen elev forteller at opplegget viste at programmering kan være gøy og finner støtte for dette i følgende utsagn fra en tredje elev:

F1 – 1: «For min del har jeg vel innsett at programmering kan være litt gøy da (...). Det at man skulle finne ut hvilken sykdom pasienten hadde – det føler jeg er en sånn typisk programmeringsoppgave vi

kunne fått mer av i biologi (...). Det var veldig gøy, og nesten litt sånn spennende faktisk (ler). Så ja, programmeringen ble litt sånn morsom da!»

To elever trekker frem elementer ved undervisningsopplegget som gjorde at de forsto programmering bedre og at de fikk en bedre oversikt over verktøyet. Samtlige av informantene i fokusgruppeintervjuet nevner imidlertid at opplegget har hatt størst effekt på holdningsdimensjonen «Syn på programmering» - gjennom å fremheve programmeringens nytteverdi i biologi. Følgende utsagn underbygger oppleggets effekt på programmeringens nytteverdi i biologi:

F1 – 2: «Jeg likte veldig godt den delen med «den skeptiske legen». Jeg hadde egentlig ikke tenkt på hvor nyttig programmering faktisk er ... før man så at det liksom var 100 000 basepar man måtte lete gjennom! (...). Det satte liksom litt perspektiv på det da!»

F2 – 1: «Jeg har jo faktisk skjønt at man kan bruke det (=programmering) i biologi også, og ikke bare i matte og sånt da. Så ja, det har faktisk endret synes ganske mye for min del!»

Flere av elevene rapporterer at opplegget hadde en effekt på evalueringen av programmering som et nyttig verktøy også inn i biologiundervisningen på skolen. To elever beskriver hvordan opplegget bidro til en bedre forståelse av temaet bioteknologi og hvordan ulike teknologier kan benyttes i behandlingen av genetiske sykdommer, slik som i følgende elevsitat:

F1 – 1: «Jeg synes det var veldig kult når man programmerte en sånn "Data-CRISPR". Det synes jeg var veldig kult, for det var liksom sånn. Jeg følte jeg på en måte skjønte det på en litt annen måte, enn når vi bare så på de tegningene i boka. Man ser jo illustrasjoner, men her så man jo det faktisk i sånn, på ... nukleotid-nivå da. Det var litt gøy!»

Eleven beskriver hvordan programmeringen bidro inn mot å forstå fagstoffet på en ny måte, ved at det ble en ny visualisering av det som skjer når gener endres. Til tross for at samtlige av elevene i intervjuene trekker frem positive aspekter ved det spesifikke undervisningsopplegget, er én av elevene usikker på hvorvidt dette ene opplegget har ført til at holdningene til å programmere i faget har endret seg. Hen kommer med følgende refleksjon:

F1 – 3: «Ja, opplegget har vist meg at programmering kan være litt gøy da. Men samtidig, så tenker jeg liksom sånn. Ok, det HER er gjort veldig nøye og sånn. Men når jeg tenker på programmering i biologi på universitetet – så blir det jo ikke helt det samme. Så selv om jeg synes dette var godt forklart og veldig

gøy, så vet jeg ikke om jeg hadde synes det samme om et annet programmeringsopplegg. Vi hadde jo en sånn annen greie tidligere, med panda og klimatall og sånn – og da skjønte jeg jo ingenting. Og da bare mista jeg helt motivasjonen ...»

Til tross for elevenes beskrivelser av denne spesifikke undervisningssituasjonen som «gøy» og «godt forklart», viser eleven til en usikkerhet om hvorvidt opplegget har ført til at holdningene til det å programmere i faget har endret seg. Hen trekker inn tidligere erfaringer mer programmering der både forståelsen og motivasjonen var lav. Eleven beskriver det slik videre i intervjuet:

F1 – 3: «Man må jo lære seg det (=programmering) for å like det. Så hvis det kommer noe annet senere og det ikke blir forklart like nøye ... så vil det jo være vanskelig igjen».

Samlet beskriver et flertall av elevene i intervjuene (n =6) at undervisningsopplegget «Code to Cure» hadde positiv effekt på de tre holdningsdimensjonene, med særlig vekt på holdningsdimensjonen «Syn på programmering» i form av nytteverdi. En av elevene beskriver positive opplevelser med opplegget, men stiller spørsmålsteget ved hvorvidt opplegget har hatt noen effekt på holdningene til å programmere i biologi.

Kapittel 6: Diskusjon

I følgende kapittel vil de presenterte resultatene drøftes opp mot teori og tidligere forskning, i et forsøk på å gjøre rede for oppgavens forskningsspørsmål. Innledningsvis følger en oppsummering av resultatene som drøftes videre i diskusjonskapitlet.

Opgavens første forskningsspørsmål (RQ1) tar sikte på å undersøke *hvilke tidligere erfaringer elevene har med programmering i skolen, og hvilken betydning erfaringene har for elevenes holdninger til å programmere i biologi*. Elevene i den aktuelle biologi 2 klassen har alle tidligere erfaringer med programmering fra videregående, flest fra faget matematikk. Elevenes grad av engasjement og mestring, er aspektene som står særlig sentralt i elevenes vurderinger av tidligere erfaringer. Elevene beskriver generelt lavt engasjement og lav mestringsfølelse ved tidligere erfaringer med programmering i undervisningen. Av faktorer som er av betydning for elevenes opplevelser, er særlig *lærers rolle og tilrettelegging og elevenes forutsetninger*, identifisert som særlig betydningsfulle. Flere av elevene har erfaringer med lærere som har lite erfaring med programmering og undervisning med programmering. Mangel på forkunnskaper innen programmering trekkes frem av samtlige av elevene i preundersøkelsen, og et flertall av elevene rapporterer å ha for få forkunnskapene til å bruke programmering inn i fagene. Resultatene viser at tidligere erfaringer med programmering i skolen er svært betydningsfulle for elevenes holdninger til å programmere i faget.

I et forsøk på å besvare oppgavens andre forskningsspørsmål (RQ2), ble *kjennetegn ved elevenes holdninger til å programmere i biologi 2* undersøkt. Resultatene viser at elevene rapporterer lav motivasjon og lavt engasjement ovenfor det å skulle programmere i faget. Andre resultater viser at elevene samlet vurderer egen kompetanse i møte med programmering som lav. Elevene hadde flere forestillinger om hva programmering er, og i hvilke situasjoner programmering som verktøy kan ha en verdi. Til tross for at flere av elevene anerkjenner verktøyets nytteverdi generelt, er de i mindre grad overbevist om programmeringens nytteverdi i biologifaget. Elevenes syn på programmering preges av forestillinger om at programmering er *for regning* og at verktøyet er *unødvendig*, da det i flere tilfeller kan erstattes av andre verktøy som fungerer like bra eller bedre. Andre forestillinger representerer elevenes syn på programmering som noe som *gjør ting mer komplisert* og som *noe ekstra* de må lære seg i fagene – i stedet for å være et verktøy som bidrar til økt læring og forståelse.

Ved oppgavens siste forskningsspørsmål (RQ3), var målet å undersøke *hvordan elevenes holdninger til å programmere i biologi 2 ble påvirket av undervisningsopplegget «Code to Cure»*. Flesteparten av elevene rapporterte at de hadde positive opplevelser med undervisningsopplegget «Code to Cure».

Oppgavens *kontekst* og valg av *rammer* og *støttestrukturer* er styrker ved opplegget som fremheves av flest elever. Av svakheter ved opplegget beskriver flere av elevene at programmeringen kunne vært mer utfordrende. Undervisningsopplegget ser ut til å ha hatt en effekt på elevenes holdninger til å programmere, og resultatene fra den parvise t-testen viste en signifikant forskjell ved 8 av de 20 påstandene ved sammenligning av pre- og post-undersøkelsen. Undersøkelsen viste til størst effekt på elevenes forestillinger og syn på programmering. En av elevene stiller seg tvilende til om «Code to Cure» har hatt en effekt på elevenes mer stabile holdninger til å programmere i faget.

6.1 Tidligere erfaringers betydning for elevenes holdninger til å programmere i biologi

Resultatene fra analysen viste at elevenes tidligere erfaringer med programmering var svært betydningsfulle for holdningene elevene har til å programmere i biologi 2. Dette gjelder erfaringer fra også andre fag enn biologi, som matematikk og fysikk. På bakgrunn av studiens funn ser ut til at elevenes affektive opplevelser ved spesifikke undervisningssituasjoner med programmering, gradvis har utviklet seg til å bli mer stabile holdninger til verktøyet generelt. Disse holdningene overfører elevene i stor grad til programmering spesifikt inn i faget biologi 2.

Flere av holdningsmodellene presentert i teorikapitlet viser til den samme sammenhengen mellom tidligere erfaringer og utviklingen av mer stabile holdninger. Som presentert i teorikapitlet, forklarer McLeod (1992) og Eagly og Chaiken (1993) begge hvordan holdninger kan oppstå på bakgrunn av tidligere affektive evalueringer eller responser mot et objekt. Elever med flere gjentatte negative opplevelser med programmering, kan etter hvert utvikle negative holdninger til å programmere mer *generelt* (Eagly & Chaiken, 1993; McLeod, 1992). Forklaringen samsvarer i stor grad med funnene fra denne studien. Eksempelvis vil elevene som har flere opplevelser med lite engasjement i møte med programmering, etter hvert kunne utvikle en negativ følelsesmessig innstilling til verktøyet og beskrive programmering som lite spennende eller kjedelig, en sammenheng som Di Martino og Zan (2010) også fant i sin studie. Flere elever beskriver tidligere erfaringer der de ikke forsto eller mestret oppgavene der de skulle programmere. Gjentatte, lignende opplevelser med manglende mestringsfølelse kan være årsaken til at flere av elevene oppfatter egen kompetanse innen programmering som lav. Inntrykket finner jeg støtte i litteraturen (Baser, 2013; Eliassen et al., 2021). Gjentatte opplevelser med programmering i såkalte «regnefag» som matematikk og fysikk, kan ha bidratt til at elevene har utviklet et syn på programmering som utelukker verktøyets nytteverdi i andre sammenhenger – slik som behandling av «big data»¹⁵ i biologifaget. Erfaringer med programmering i situasjoner der andre verktøy

¹⁵ Big data i biologi refererer her til store og komplekse datasett som inneholder informasjon om biologiske systemer og fenomener – som for eksempel DNA-sekvenser. Stephens, Z. D., Lee, S. Y., Faghri, F., Campbell, R. H., Zhai, C., Efron, M. J., Iyer, R., Schatz, M. C., Sinha, S. & Robinson, G. E. (2015). Big data: astronomical or genomics? *PLoS biology*, 13(7), e1002195.

ville fungert like bra – eller bedre, kan ha påvirket elevenes holdninger og fremmet et syn på programmering som lite nyttig eller unødvendig.

Enkelte av elevene viser til tilfeller der de har hatt positive erfaringer med programmering, men likevel uttrykker mindre positive holdninger til programmering generelt. Dette forholdet kan komme av flere årsaker, der de aktuelle elevene kan ha *flere* eller *sterkere* negative opplevelser med programmering, og at disse opplevelsene «overstyrer» de positive, slik at mer negative holdninger til verktøyet formes. Dette understreker viktigheten av at elevene får positive opplevelser med programmering i undervisningen, slik at elevenes holdninger til læringsverktøyet gradvis kan stimuleres (Andersen, 2022; Du et al., 2016; Eliassen et al., 2021; Erol, 2020; Forrester et al., 2022; McLeod, 1992; Vinnervik, 2021).

I denne studien viste det seg at særlig to aspekter står sentralt i elevenes vurderinger av tidligere erfaringer som positive eller negative, nemlig grad av engasjement og grad av mestring. I en svært stor andel av tilfellene der elevene beskrev negative opplevelser knyttet til undervisningssituasjoner med programmering, skyldes dette i stor grad mangel på mestring. Dette funnet finner støtte flere steder i forskningslitteraturen (Andersen, 2022; Eliassen et al., 2021; Sengupta et al., 2013; Vinnervik, 2021; Zukanovic, 2021).

Flere faktorer ble avgjørende for elevenes erfaringer, og særlig faktorene *lærerens rolle og tilrettelegging*, og *elevenes forutsetninger* ble fremhevet i elevenes beskrivelser. Flere av elevene har erfaringer med lærere som selv kan lite programmering, og som mangler kompetanser i hvordan implementere programmering i undervisningen. I tilfellene der elevene opplever at læreren ikke har nok kompetanse, resulterer dette ofte i at elevene ikke får tilfredsstillende støtte og veiledning når de står fast med programmeringen. Utfordringene som oppstår på bakgrunn av lærerens manglende programmeringstekniske og programmeringsdidaktiske kompetanse, beskrives av både Munthe (2022) og Vinnervik (2021). Forfatterne fremhever betydningen av lærerens kompetanse på valg av rammer og støttestrukturer – som arbeidsmåter, nivå og tema. Til tross for at dette ikke er fokuset for denne oppgaven, er det viktig å anerkjenne de utfordringene lærerne står ovenfor ved innføringen av programmering inn i skolefagene (Munthe, 2022; Vinnervik, 2021), og betydningen dette har for elevenes opplevelser av mestring og engasjement (Eliassen et al., 2021; Erol, 2020; Haraldsrud et al., 2020).

Et flertall av elevene i studien beskriver hvordan *mangelen på egne forkunnskaper* innen programmering har vært en avgjørende faktor i tilfellene der de skulle programmere tidligere. Elevene viser til flere tilfeller der nivået på oppgaven eller undervisningen har vært på et for høyt nivå eller har tatt utgangspunkt i at elevene kan mer enn det de faktisk kan. Dette kan bidra til at elevene opplever lite

mestring i møte med programmeringsaktiviteter (Eliassen et al., 2021), og resultatet var til en viss grad forventet, da tidligere studier har vist at de fleste elever har få forkunnskaper i programmering (Eliassen et al., 2021; Munthe, 2022; Zukanovic, 2021). Elevene i studien viser til et behov for å forstå det grunnleggende innen programmering, *før* de blir trygge på å bruke programmering inn i de allerede etablerte fagene. Dette krever tid, og blir trolig ikke alltid prioritert når undervisningen planlegges. Når elevene opplever nivået for høyt og ikke får tilfredsstillende oppfølging underveis, ofte på bakgrunn av at læreren ikke kan hjelpe alle samtidig, viser studiens funn at dette kan resultere i at elevene ikke kommer seg videre, ikke forstår oppgaven eller inntar en mindre aktiv læringsstrategi – som å kopiere kode uten å forstå hva koden gjør og hvorfor den er bygget opp på denne måten den er.

Til tross for at lærernes og elevenes forutsetninger for å programmere i fagene trolig vil forandre seg i årene som kommer (Munthe, 2022), støtter resultatene fra denne studien det tidligere forskning fremhever av utfordringer når programmering integreres i skolefagene. Flere av utfordringene kan knyttes opp mot lærernes mangel på programmeringsteknisk og programmeringsdidaktiske kompetanse (Munthe, 2022; Vinnervik, 2021) og elevenes manglende forkunnskaper i programmering (Eliassen et al., 2021; Munthe, 2022; Zukanovic, 2021). Dette viser at flere av de tidligere presenterte utfordringene fra ulike fag, land og utdanningsnivå, også i stor grad er gjeldene i denne konteksten, som er biologi 2 på videregående nivå i Norge. Utfordringene viser seg å være avgjørende for elevenes evalueringer av tidligere opplevelser med programmering i undervisningen, som igjen er av betydning når elevenes holdninger til læringsverktøyet formes – både generelt og i faget biologi 2.

6.2 Elevenes holdninger til å programmere i biologi 2

Med utgangspunkt i definisjonen presentert i teorikapitlet, ble tilfellene der elevene beskrev responser av *moderat* intensitet og stabilitet ovenfor programmering, tolket som uttrykk for elevenes *holdninger til å programmere*. Responsene kom til uttrykk der elevene viste til mer «generelle» eller «overordnede» responser, som følelser, oppfatninger og forestillinger til verktøyet, *uten* å referere til spesifikke undervisningssituasjoner med programmering. I denne studien fremkom det at elevenes mer stabile responser ofte kom til uttrykk ved minst én av de tre holdningsdimensjonene beskrevet av Di Martino og Zan (2010); følelsesmessig innstilling, oppfatning av egen kompetanse og syn på programmering. Analysen viste at elevene i stor grad har samme holdninger til programmering generelt, som spesifikt inn i faget biologi 2. I diskusjonen vil jeg omtale disse som «holdninger til å programmere», uten å skille mellom de to – med unntak av enkelte tilfeller der det spesifisert. Merk at elevenes holdninger slik de beskrives her, er fremstilt med utgangspunkt i datamaterialet fra *pre-undersøkelsen*.

Generelt viste resultatene at elevene hadde en *negativ følelsesmessig innstilling* til det å programmere i faget. Ofte kom dette til uttrykk ved utsagn som «jeg liker ikke programmering», og oppfatningen støttes av elevenes graderinger av påstandene i tabell 11. I sine begrunnelser viser elevene ofte til beskrivelser om programmering som vanskelig og noe de ikke mestrer, og flere av elevene beskriver programmering som lite engasjerende og lite motiverende. Disse resultatene samsvarer med funn fra tidligere studier, som viser at elever og studenter ofte opplever programmering som lite engasjerende (Baser, 2013; Forrester et al., 2022), lite interessant (Du et al., 2016; Eliassen et al., 2021) og lite motiverende (Andersen, 2022; Eliassen et al., 2021). Forrester et al. (2022) finner at flere av studentene ofte er bekymret og urolige når de skal programmere, og refererer til disse følelsene som «programmeringsrelatert angst» (min oversettelse). Elevenes gradering av påstand A5 i tabell 11 ligger så vidt over det nøytrale midtpunktet, og det er derfor usikkert hvorvidt en slik programmeringsrelatert angst er representert blant elevene i denne studien. På bakgrunn av oppgavens begrensede omfang velger jeg derfor å ikke gå videre inn på dette i diskusjonen. Det ser ut til at holdningsdimensjonen «følelsesmessig innstilling» i stor grad sammenfaller med de to prosessene beskrevet av modellen til Hannula (2002) som omhandler følelsene elevene får når de jobber med matematikk, og følelsene elevene automatisk forbinder med konseptet matematikk. Dette styrker antagelsen om at holdningsdimensjonen «følelsesmessig innstilling» er viktig i elevenes beskrivelser av hvilke holdninger til de har til læringsverktøyet.

Elevenes oppfatning av egen kompetanse i møte med programmering, utgjør den andre holdningsdimensjonen som ble avgjørende i elevenes beskrivelser. Resultatene viser at flertallet av elevene i studien oppfatter egen kompetanse i møte med programmeringsaktiviteter som *lav*. Dette kom til uttrykk ved at elevene beskrev *få* tidligere mestringsopplevelser med programmering fra undervisning, og liten tro på egne ferdigheter i fremtidige møter med læringsverktøyet. Resultatet støttes av graderingene i tabell 13. I besvarelsene beskriver et flertall av elevene programmering som *noe vanskelig*. Tidligere forskning har vist at elever ofte har få forkunnskaper i programmering (Eliassen et al., 2021; Munthe, 2022; Zukanovic, 2021). Eliassen et al. (2021) fant i sin studie at biologistudentene ofte hadde lav mestringsforventning i emner som integrerer programmering. På bakgrunn av dette, var funnene i denne studien til en viss grad forventet. Nesten samtlige av elevene begrunner sin negative innstilling til at programmering skal inn i biologi 2 undervisningen, med utsagn som at programmering er vanskelig og noe de ikke mestrer. Dette indikerer at elevenes oppfatning av egen kompetanse i møte med programmering, er svært avgjørende for deres holdninger. Inntrykket støttes av Baser (2013), som i sin studie fant at det var en sentral sammenheng mellom studentenes opplevelse av mestring når de programmerer, og holdningene de har til verktøyet. Lav mestringsfølelse kan bidra til at elevene kan utvikle lav mestringsforventning (Eliassen et al., 2021). Dette kan virke inn på senere kontekster der

elevene skal programmere i fag, og fremhever viktigheten av at det legges til rette for positive mestringsopplevelser hos elevene når de programmerer i undervisningen.

I denne studien viste det seg at flere *forestillinger* preger elevenes syn på hva programmering er, og i hvilke tilfeller programmering eller kompetanser innen programmering er nyttig. De mest gjennomgående forestillingene kan kobles til spørsmål om verktøyets nytteverdi. Flere av elevene anerkjenner programmeringens nytteverdi generelt, men færre ser ut til å se nytteverdien av verktøyet inn i biologi 2. Dette inntrykket støttes av elevenes graderinger presentert i tabell 14. Flere av elevenes forestillinger bærer preg av at elevene har flest erfaringer med programmering fra fag som matematikk og fysikk, og dette kan forklare hvorfor flere av elevene vurderer programmering som mindre relevant inn mot faget biologi. Det kan også forklare hvordan flest av elevenes eksempler på når programmering kan være nyttig i faget, er koblet opp til modellering og beregninger. Enkelte av elevenes har et syn på programmering som *unødvendig*, og begrunner dette med at de i flere tilfeller har brukt programmering i undervisningen der andre verktøy kunne fungert like bra – eller bedre. Denne oppfatningen introduserer et interessant perspektiv, og presenterer muligheten for at elevene har tidligere tilnærminger med programmering fra undervisningen der læringsverktøyet ikke har vært integrert der det er hensiktsmessig. Det kan også skyldes at elevene nå er på et så «grunnleggende» programmeringsteknisk nivå, at det er vanskelig å utvikle undervisningsopplegg som tar høyde for elevenes forutsetninger, samtidig som det fremhever verktøyets verdi i fagene. Dette perspektivet illustreres godt av elev SS1-7 i pre-undersøkelsen. Munthe (2022) og Sengupta et al. (2013) understreker viktigheten av at programmeringen integreres i fagene der det er hensiktsmessig, slik at faget er i fokus og ved at verktøyet bidrar til økt forståelse av det *faglige*. Dette blir i større grad en utfordrende oppgave dersom elevene ikke har forkunnskapene som kreves, eller ved at tiden i klasserommene må brukes på å lære elevene det grunnleggende i programmering, før de i hele tatt kommer i gang med oppgaven.

Di Martino og Zan (2010) introduserer ideen om at de tre holdningsdimensjonene henger sammen og påvirker hverandre gjensidig. Basert på elevenes beskrivelser, er det naturlig å anta at det eksisterer flere slike sammenhenger også i denne konteksten. Særlig sammenhenger mellom dimensjonene *følelsesmessig innstilling og oppfatning av egen kompetanse*, trekkes frem som sentral hos elevene. Et flertall av elevene begrunner en negativ følelsesmessig innstilling i form av lavt engasjement og motivasjon, med mangelen på tidligere mestringsopplevelser. En lignende sammenheng introduseres også av Andersen (2022), som fant en tydelig sammenheng mellom grad av mestring på oppgaver, og følelsene elevene får til programmering i matematikk. Flere av elevene gir uttrykk for at de «ikke liker programmering», fordi de ikke kan nok, og denne koblingen finner støtte flere steder i litteraturen (Baser, 2013; Eliassen et al., 2021; Forrester et al. 2022).

Di Martino og Zan (2010) hevder at holdningsdimensjonen *følelsesmessig innstilling*, henger tett sammen med de to andre dimensjonene, ved at en negativ følelsesmessig innstilling ofte oppstår på bakgrunn av *liten tro på egen kompetanse* eller et *negativt syn på programmering*. Et annet forhold som kan være relevant, er derfor sammenhengen mellom elevenes syn på programmering og elevenes motivasjon for å programmere. Tidligere forskning har vist til viktigheten av å fremheve programmerings nytteverdi for å stimulere en slik motivasjonen (Andersen, 2022; Pevzner & Shamir, 2009; Sengupta et al., 2013; Vinnervik, 2021). Funn fra denne studien viser at elevene som ikke ser nytteverdien av programmering i faget – for eksempel gjennom forestillingene om at programmering er unødvendig, at programmering gjør ting mer komplisert, eller programmering «blir noe ekstra», viste lavere motivasjon til å programmere i faget. Dersom elevene ikke ser verdien eller relevansen av læringsverktøyet, er sannsynligheten stor for at elevene blir mindre motiverte for å lære programmering, og på å bruke tid på verktøyet i undervisningen.

6.3 Erfaringer med undervisningsopplegget «Code to Cure»

6.3.1 Elevenes opplevelser med undervisningsopplegget

Funn fra studien viste at elevene i all hovedsak hadde positive opplevelser med undervisningsopplegget «Code to Cure». Inntrykket baserer seg på egen og faglærers observasjoner ved gjennomføringen, samt elevenes mer generelle meninger om undervisningsopplegget. Basert på elevenes graderinger i etterkant av undervisningsøkta (tabell 16), viste samtlige av elevene at de likte undervisningsopplegget. Det var få spørsmål til meg og faglærer, og alle gruppene fullførte hoveddelen av opplegget. Dette underbygger oppfatningen av at elevene opplevde mestring i arbeidet med aktiviteten, som videre støttes av elevenes graderinger av påstandene i tabell 16 og 17. Flere av elevene rapporterer at opplegget førte til at de lærte noe (A4 i tabell 16), og analysen av elevens åpne tekstsvaer, viser at elevene beskriver undervisningsopplegget som spennende, interessant og motiverende. Resultatene tyder på at elevene generelt hadde positiv opplevelse med undervisningsopplegget «Code to Cure».

Flere faktorer ved «Code to Cure» ble avgjørende for elevenes opplevelser med undervisningsopplegget. I metodekapitlet ble prosessen med utviklingen av undervisningsopplegget beskrevet, som viser at «Code to Cure» ble utviklet med utgangspunkt i tidligere forskning, samt elevenes holdninger til å programmere i biologi. Basert på elevenes evalueringer ser det ut til at flere av de tilpasningene og valgene som ble tatt i prosessen, påvirket elevenes opplevelser positivt. Disse faktorene trekkes frem som *styrker ved undervisningsopplegget*, og i denne delen av diskusjonen fremheve to av de styrkene som trekkes frem som mest betydningsfulle for elevenes opplevelser.

Det viste seg at oppgavens *kontekst* var av særlig stor betydning i elevenes evalueringer av undervisningsopplegget. Ifølge tidligere forskning kan det å plassere programmeringen i en kontekst som interesserer elevene, være en god måte å integrere programmering inn i undervisningen av andre fag (Andersen, 2022; Munthe, 2022; Pevzner & Shamir, 2009; Sengupta et al., 2013). Oppgavens kontekst bidro til at elevene opplevde økt motivasjon og engasjement. Flere elever hevder at konteksten bidro til at programmeringen fikk *et tydelig mål*, og at dette igjen resulterte til at elevene ble investert i å fullføre oppgaven. Konteksten ble ifølge elevene også viktig for å fremheve programmeringens nytteverdi i biologi – her i temaer innenfor bioteknologi. Særlig oppgaven med den skeptiske legen ble av elevene vurdert som avgjørende i bevisstgjøringen av verktøyets nytteverdi. Flere gir uttrykk for at oppgaven bidro inn med et viktig perspektiv for å illustrere hvor enormt tidkrevende og unøyaktig gentesting ville vært uten hjelp av programmering.

Basert på elevenes tidligere erfaringer og holdninger, ble det tatt valg om undervisningsoppleggets rammer og støttestrukturer. Flere av elevene beskriver tidligere erfaringer med programmering der oppgaven var vanskelig å forstå, eller tilfeller der de følte at de ikke fikk den hjelpen de trengte underveis i arbeidet. Ved å inkludere grundige forklaringer med eksempler, hint og løsningsforslag, var tanken at elevene skulle trygges underveis i arbeidet (Haraldsrud et al., 2020), og at det ville løsrive meg og faglærer slik at vi fikk færre spørsmål til det programmeringstekniske, og kunne fokusere på å hjelpe elevene med spørsmål til det faglige (Forrester et al., 2022). Funnene fra studien viser at elevene generelt opplevde disse støttestrukturene som positive og svært betydningsfulle for deres opplevelser med undervisningsopplegget. Støttestrukturene bidro til at elevene ble mer selvstendige, og gjorde det mulig å tilpasse opplegget til elevenes ulike nivå (Forrester et al., 2022; Haraldsrud et al., 2020), ved at de selv kunne velge om de ville bruke hint og løsningsforslag. Et flertall av elevene trekker frem at de likte oppsettet med nettsiden, de ulike rommene og læringssti-heftet (vedlegg B). I en artikkel av Guzman et al. (2019) presenteres ideen om at man kan redusere belastningen på elever som lærer seg programmering, ved å redusere antall elementer i et læringsmateriale og bryte ned og dele opp komplekse oppgaver i mindre, håndterbare trinn. Tilpasningene kan bidra til at elevene får en mer positiv læringsopplevelse, i form av økt motivasjon, mindre frustrasjon og mindre stress. I fokusgruppeintervjuene er samtlige av elevene enige i at de likte hvordan oppsettet med nettsiden fordelte og brøt opp de ulike programmeringsoppgavene, slik at de ikke var nødt til å se alle programmeringsoppgavene samtidig. Inndeling kan har vært med på å gjøre omfanget av opplegget og programmeringen mindre overveldende for elevene, og trolig økte sannsynligheten for at de ikke «ga opp» før de i hele tatt kom i gang med aktiviteten.

Også andre elementer ved undervisningsopplegget fremheves av elevene som styrker ved undervisningsopplegget. På bakgrunn av oppgavens omfang, vil det ikke være mulig å diskutere alle faktorene her. Jeg ønsker likevel å anerkjenne hvordan faktorene nivå, variasjon og samarbeid trekkes frem som viktige for flere av elevenes opplevelser. Elevenes evalueringer i etterkant av gjennomføringen (tabell 17), viser at elevene i stor grad opplevde *nivået* på programmeringen som passende, og fremhever dette som en av styrkene ved opplegget. Dette virker rimelig ut fra flere av anbefalingene i litteraturen (Munthe, 2022; Pevzner & Shamir, 2009). Flere av elevene vurderte *variasjonen* i undervisningsopplegget som positiv, og beskriver hvordan oppgavene som ikke handlet om programmering bidro til at de fikk et «avbrekk» underveis, og ble mer investert og utholdende i møte med oppgavene der de skulle programmere. Et flertall av elevene evaluerte samarbeidet som en av styrkene ved undervisningsopplegget, og det finnes mye litteratur som fremhever fordelene ved å la elevene jobbe sammen i aktiviteter der de skal programmere (Du et al., 2016; Forrester et al., 2022; Munthe, 2022).

Andre faktorer ble av elevene vurdert som *svakheter ved undervisningsopplegget*, og den mest fremtredende faktoren handler om at flere av elevene opplevde nivået på programmeringen som for lite utfordrende. Enkelte av elevene gir uttrykk for at flere av hintene og forklaringene underveis, med fordel kunne vært droppet eller skjult, slik at elevene i større grad fikk mulighet til å utforske og tenke selv. Dette var et noe forventet funn, da undervisningsopplegget bevisst ble utviklet slik at det ville kreve få forkunnskaper for elevene. Det viser likevel til viktigheten av å differensiere oppgaver med programmering slik at flest mulig opplever mestring, samtidig som det sørger for at det er nok utfordringer til elevene som kan mer (Haraldsrud et al., 2020). Til tross for dette, fremhever flertallet av elevene nivået på programmeringen og støttestrukturene som styrker ved undervisningsopplegget. Det kan derfor virke som at elevene her er noe splittet i sine evalueringer. En mulig årsak til denne delingen kan være elevenes ulike nivå og forkunnskaper. Dette understreker verdien av å utvikle undervisningsopplegg som kan tilpasses elevenes ulike nivå og ulike behov for støtte (Haraldsrud et al., 2020; Munthe, 2022; Sengupta et al., 2013). Andre forslag til forbedringer ved undervisningsopplegget handlet om konkrete aspekter ved nettsiden, eller feil i koden som representerte en av pasientenes DNA.

Til tross for at elevene fremhevet flere svakheter ved undervisningsopplegget, er det lite som indikerer at de mindre positive faktorene ble avgjørende før elevenes helhetlige opplevelse med «Code to Cure». Dette inntrykket baserer seg på resultatene om elevenes mer generelle opplevelser ved gjennomføringen som viste at elevene generelt hadde positive opplevelser med opplegget. En mulig

antagelse er at svakhetene ikke var «avgjørende nok», eller at elevene opplevde styrkene som «større» enn svakhetene, og på denne måten fortsatt satt igjen med et mer positivt helhetsinntrykk.

6.3.3 Undervisningsoppleggets effekt på elevenes holdninger til å programmere i biologi 2

Et av de mest sentrale spørsmålene for denne oppgaven, handler om hvordan undervisningsopplegget «Code to Cure» påvirket elevenes holdninger til å programmere i faget biologi 2. Flere av målingene og funn fra analysen viser til at opplegget hadde en effekt på elevenes holdninger til å programmere i faget. Med utgangspunkt i det kvantitative datamaterialet, viste analysen en forskjell i gjennomsnitt før og etter gjennomføringen på åtte av tjue påstander. Sammen med resultatene fra den kvalitative analysen, kan resultatene si noe om elevenes opplevelser med opplegget og påvirkningen det kan ha hatt på elevenes holdninger til å programmere i faget.

Resultatene viste til det som ble definert som «ønsket effekt» på elevenes følelsesmessige innstilling til å programmere generelt, og i faget biologi 2. Effekten kom til uttrykk i elevenes graderinger av påstandene i tabell 20, og analysen av det kvalitative datamaterialet. Samlet viste studiens funn at elevene i større grad rapporterte motivasjon og engasjement ovenfor det å programmere i undervisningen. Enkelte av elevene fremhever hvordan undervisningsopplegget har vist at programmering kan være morsomt (tabell 21), et funn som står i kontrast til elevenes beskrivelser fra pre-undersøkelsen. Tidligere forskning har vist at elevenes interesse, engasjement og motivasjon kan øke på bakgrunn av valg av undervisningsopplegg (Erol, 2020; Sengupta et al., 2013), slik som i denne studien. Sengupta et al. (2013) fremhever oppgavens *kontekst* som særlig avgjørende for en slik endring. Dette er et funn som viser seg å være gjeldende også i denne konteksten, da flere av elevene fremhever «historien bak» som avgjørende for deres evalueringer av opplegget som *spennende og interessant*.

Fire av påstandene i tabell 20 kan kobles til holdningsdimensjonen «oppfatning av egen kompetanse», og viser til at elevene ved pre-undersøkelsen i større grad har tro på egen mestring i situasjoner der de programmerer. Forskjellene i gjennomsnitt viser til at elevene i mindre grad evaluerer programmering som noe «vanskelig», og at elevene er mer komfortable med å bruke programmering i undervisningen. Basert på de tidligere presenterte resultatene som viste at elevene i all hovedsak hadde positive mestringsopplevelser med «Code to Cure», gir det mening dersom opplegget har hatt en positiv effekt på også denne holdningsdimensjonen.

Likevel ønsker jeg i lys av min egne helhetlige opplevelse og etterfølgende analyse, å stille spørsmålsteget ved hvorvidt opplegget faktisk hadde en effekt på elevenes oppfatning av egen kompetanse. En av

elevene i fokusgruppeintervjuet (F1-3) stilte seg tvilende til om opplegget var nok til at holdningene til å programmere i faget var endret. Dette ble begrunnet med at eleven hadde lite tro på at fremtidige opplegg kom til å være utformet på samme måte med tanke på nivå, støttestrukturer og gjennomføring. Eleven ga uttrykk for at hen trolig ville synes programmering er like vanskelig som før, neste gang hen skulle programmere i et annet fag på skolen eller videre studier. Dette sammenfaller med tidligere forskning, som viser at elevene ofte møter på flere utfordringer når programmering og fag kombineres (Munthe, 2022; Sengupta et al., 2013; Waters, 2020). Dersom elevene har lite tro på at fremtidige undervisningsopplegg vil føre til lignende positive mestringsfølelser, kan dette trolig virke inn på elevenes *mestringsforventning* (Eliassen et al., 2021). Dette viser til lignende resultater som i studien til Du et al. (2016), som *ikke* fant en signifikant forskjell på elevenes ferdigheter i programmering etter et enkelt undervisningsopplegg. Til tross for dette viser resultatene til Erol (2020) og Forrester et al. (2022) at elevene/studentenes selvtillit og selvoppfatning økte som resultat av et bestemt undervisningsopplegg. Gjennomføringene på de disse to prosjektene varte i henholdsvis 12 uker og 3 semestre. Tid er derfor et faktor som skiller disse gjennomføringene, fra min egen og den til Du et al. (2016). I lys av litteratur er det derfor trolig for mye å håpe på, at ett enkelt undervisningsopplegg med programmering er nok til å påvirke elevenes oppfatning av egen kompetanse i møte med programmering. For å oppnå en slik endring, vil elevene trolig trenge flere positive mestringsopplevelser med verktøyet (Du et al., 2016; McLeod, 1992), gjerne over lengre tid.

Undervisningsopplegget «Code to Cure» hadde trolig størst effekt på holdningsdimensjonen «Syn på programmering». Elevenes graderinger av påstandene i tabell 19 viser at elevene i all hovedsak forsto hvorfor programmeringen var en del av undervisningsopplegget, og at programmering er viktig for analyse av store mengder data i biologi. Oppfatningen støttes av resultatene fra den parvise t-testen (tabell 20), som viste en signifikant effekt på påstandene vedrørende verktøyets nytteverdi i faget. I det kvalitative datamaterialet kom effekten til uttrykk ved at elevene viste til en økt bevisstgjøring omkring programmeringens nytteverdi i biologi, men også som verktøy inn i undervisningsfaget biologi 2. Her ble trolig oppleggets kontekst og rammer avgjørende, i form av at elevene ble introdusert for et reelt tilfelle der programmering er nyttig og verdifullt i biologi (Andersen, 2022; Munthe, 2022; Pevzner & Shamir, 2009; Sengupta et al., 2013). I denne konteksten er det kanskje vanskeligere for elevene å se for seg andre verktøy som ville fungert like godt eller bedre, noe som kan ha bidratt til at færre elever opplevde programmeringen som «unødvendig» - slik det ble beskrevet av flere i pre-undersøkelsen. Funnene sammenfaller med resultatene fra studien til Du et al. (2016), som fant at undervisningsopplegget hadde en positiv effekt på elevenes holdninger til programmering ved at det tydeliggjorde programmeringens nytteverdi i faget.

Oppleggets effekt på elevenes syn på programmering, kom også til uttrykk ved elevenes forestillinger om programmering som verdifullt inn i *undervisningen* av biologi. I pre-undersøkelsen kom det frem at flere av elevene opplevde programmeringen som noe *ekstra*, og noe de måtte lære seg i tillegg til alt det andre i faget. Etter gjennomføringen hevder flere av elevene at undervisningsopplegget førte til at de forsto flere av de biologifaglige konseptene bedre. Sitatet fra en av elevene (F1-1) presenterer ideen om at programmeringen ble en annen måte å visualisere det som skjer når gener endres ved hjelp av CRISPR, og at dette bidro til at eleven fikk en bedre forståelse av prosessene innen genteknologi. Tanken om at programmering kan bidra positivt inn i undervisningen i form av økt forståelse i fagene, finner vi igjen i litteraturen (Haraldsrud et al., 2020; Munthe, 2022; Sengupta et al., 2013; Vinnervik, 2021; Waters, 2020). Munthe (2022) presiserer i sin doktorgradsavhandling at programmeringen skal virke *for faget*, og ikke omvendt. For flere av elevene ble programmeringen i «Code to Cure» en ny representasjonsform av fagstoffet. Dette kan ha bidratt til at elevene fikk nye tilnærminger til faget, og at de på den måten forsto de biologifaglige konseptene på en ny måte. Trolig kan dette ha bidratt til at flere av elevene fikk et nytt syn på programmering i faget, ved å gå fra å være «noe ekstra», til å bli et verktøy som kan bidra inn mot å forstå faget og biologifaglige konsepter bedre.

Di Martino og Zan (2010) presenterer ideen om at man ved å påvirke én av de tre holdningsdimensjonene, indirekte kan påvirke de to andre. Dette henger sammen med forfatterens argument om at de tre holdningsdimensjonene henger tett sammen og påvirker hverandre gjensidig, slik det er beskrevet tidligere i diskusjonskapitlet. Analysen viser at en slik påvirkning trolig er gjeldene også i denne studien, der elevenes beskrivelser viser til at undervisningsoppleggets effekt på én dimensjon, førte til en effekt på de to andre dimensjonene. Her vil jeg presentere tre av sammenhengene som var mest fremtredende i elevenes evalueringer:

Elevenes motivasjon for å programmere i biologi 2, så ut til å øke når elevene ble bevisstgjort verktøyets nytteverdi i faget og i undervisningen. Dette stemmer overens med tidligere forskning, som viser at det er en sammenheng mellom elevenes syn på programmering og motivasjon for å programmere (Andersen, 2022; Pevzner & Shamir, 2009; Sengupta et al., 2013; Vinnervik, 2021). Trolig kan flere av elevene oppleve det som mer motiverende dersom de forstår at programmeringen har en verdi i faget. Motivasjonen kan stimuleres ytterligere dersom elevene ser verdien av læringsverktøyet for å *forstå faget*, slik det er beskrevet tidligere i diskusjonen.

En annen relevant sammenheng ble påvist mellom *elevenes grad av motivasjon som følge av opplevd mestring*. Studien viser at elevenes motivasjon økte som resultat av at elevene opplevde mestring i møte med programmeringsaktiviteten. Dette resultatet var til en viss grad forventet, da tidligere forskning har

vist at det er en tydelig sammenheng mellom grad av mestring og elevers motivasjon (Andersen, 2022; Baser, 2013; Eliassen et al., 2021; Forrester et al., 2022). Denne sammenhengene understreker igjen viktigheten av at elevene opplever undervisning med programmering som kan gi positive mestringsopplevelser.

En tredje sammenheng kobler alle de tre holdningsdimensjonene sammen, og kom til uttrykk der elevene viste til at *en økt bevisstgjøring om verktøyets nytteverdi*, førte til at de ble *mer motivert for å programmere*, som igjen ble avgjørende for enkelte av elevenes *utholdenhet og mestring*. Eliassen et al. (2021) hevder at elever som er mindre motiverte og viser mindre interesse for et fag, er mer sårbare dersom de møter motgang underveis i semesteret. En slik motgang kan være uheldig for elevenes læringsprosesser, og forfatterne argumenterer derfor for at man bør videreutvikle nytteverdien studentene ser i emnet. Med utgangspunkt i dette argumentet, er det grunn for å tro at det samme gjelder den andre veien. Dersom en bevisstgjøring av nytteverdi bidro til at elevene ble mer motiverte, kan dette ha bidratt til at elevene ble mer utholdende i arbeidet med aktiviteten. Denne utholdenheten kan ha vært særlig avgjørende for de elevene som har få tidligere mestringsopplevelser med programmeringer fra tidligere, og kan ha bidratt til at elevene fullførte aktiviteten til tross for eventuell motgang. På denne måten kan programmeringens nytteverdi ha bidratt til at flere av elevene fikk positive mestringsopplevelser med «Code to Cure».

Sammenhengene er hentet fra elevenes beskrivelser, og utgjør eksempler som illustrerer hvordan de tre dimensjonene henger sammen og gjensidig påvirker hverandre. Sammenhengene introduserer teorier om at man ved hjelp av forholdsvis «få» grep, i form av valg av undervisningsopplegg, kan påvirke flere av dimensjonene som er avgjørende for elevenes holdninger til å programmere i fagene, samtidig.

Betraktninger om oppleggets effekt på elevenes holdninger

Til tross for flere indikasjoner på at «Code to Cure» hadde en effekt på elevenes holdninger til å programmere i biologi 2, ønsker jeg avslutningsvis å inkludere noen perspektiver som kan være viktige for å forstå hvordan vi skal tolke denne effekten. McLeod (1992) og Eagly og Chaiken (1993) skiller mellom elevenes mer intense, følelsesmessige reaksjoner og elevenes mer stabile holdninger. Basert på resultatene, har flere av elevene trolig hatt en positiv «her og nå» opplevelse med undervisningsopplegget. Dette er tilsvarende det McLeod (1992) beskriver som mer intense, mindre stabile følelser eller følelsesmessige reaksjoner. For å forstå resultatene, blir det viktig å ikke sette likhetstegn mellom det at elevene fremsto entusiastiske og motiverte rett i etterkant av undervisningsopplegget, med det at elevene er entusiastiske og motiverte for å programmere i faget

generelt. Tidligere forskning viser at elevers holdninger gradvis dannes ved gjentatte, lignende opplevelser (McLeod, 1992). Dersom elevene har flere negative mestringsopplevelser med programmering i fagene, er det naturlig å anta at det skal mer til enn ett enkelt undervisningsopplegg for å endre disse, og trolig trenger elevene flere, positive opplevelser med læringsverktøyet. Det blir derfor viktig å ikke bagatellisere de utfordringene det kan være med å endre elevers holdninger til et verktøy som programmering.

Et annet viktig poeng handler om at elevene i den aktuelle biologiklassen trolig er klar over at fremtidige undervisningsopplegg med programmering ikke vil bli lagt opp på samme måte som «Code to Cure». Flere av elevene vet sannsynligvis at dette er et undervisningsopplegg som har tatt utgangspunkt i nettopp *deres* forutsetninger og holdninger, og designet helt spesifikt for at de skal få en positiv opplevelse med programmering i faget. Flere av elevene er nok også klar over at det har tatt tid å utvikle «Code to Cure», og at de aller færreste lærere har mulighet til å sette av denne tiden for ett enkelt undervisningsopplegg (Haraldsrud et al., 2020; Vinnervik, 2021). Samlet kan dette være årsaker til at eleven fra fokusgruppeintervjuet fortsatt er tvilende til om holdningene hen har til å programmere i biologi og andre fag, har blitt endret av dette undervisningsopplegget alene, og at effekten på holdningsdimensjonen «Oppfatning av egen kompetanse» er den minst fremtredende. For å øke elevenes oppfatning av egen kompetanse, vil elevene trolig trenge flere positive mestringsopplevelser med verktøyet over lengre tid (Du et al., 2016; Erol, 2020; Forrester et al., 2022; McLeod, 1992).

Til tross for at det finnes flere perspektiver det er viktig å ta hensyn til når studiens resultater diskuteres, viser flere av studiens funn til at undervisningsopplegget «Code to Cure» faktisk hadde en effekt på elevenes holdninger til å programmere i faget. Resultatene åpner opp for flere implikasjoner for lærere og skolen, som jeg vil komme tilbake til i oppgavens siste del.

6.4 Studiens styrker og begrensninger

I metodekapitlet ble de valgene som er tatt med hensyn til reliabilitet og validitet for å styrke studiens overordnede kvalitet, presentert. I dette avsnittet vil jeg drøfte hvordan studiens styrker og begrensninger kan ha påvirket oppgavens *gyldighet* og *generaliserbarhet*. Jeg vil innledningsvis diskutere noen av elementene knyttet til spørreskjemaene, fokusgruppeintervjuene og den parvise t-testen hva gjelder gyldighet. Avslutningsvis vil jeg gjøre rede for studiens generaliserbarhet og drøfte hvorvidt resultatene kan ha en overføringsverdi til andre elever i lignende situasjoner.

Flere av påstandene i spørreskjemaet inkluderer begreper som kan være vanskelige å måle, slik som begrepet *motivasjon*. Til tross for at jeg i utarbeidingen av spørreskjemaene forsøkte å veie opp for disse utfordringene ved å inkludere flere påstander for samme teoretiske begrep, er det er sannsynlighet for at elevene kan ha tolket begrepene annerledes enn det det var ment. Dette igjen kan ha påvirket resultatene og studiens gyldighet (Frønes & Pettersen, 2021). Ved at jeg i intervjuene kunne stille oppklarende spørsmål til elevene, fikk jeg i større grad kartlagt hva elevene la i de ulike begrepene som motivasjon, mestring og nytteverdi. Til tross for at det alltid vil være en risiko for at målingene ikke lykkes å måle de teoretiske begrepene de er ment å måle (Frønes & Pettersen, 2021), vil forhåpentligvis metodetrianguleringen, og den nøye utarbeidingen av spørreskjema og intervjuguide likevel ha bidratt med å styrke korrespondansen mellom begrepene og de fenomenene jeg ønsket å måle (Frønes & Pettersen, 2021).

I fokusgruppeintervjuene er det flere elementer som kan sies å påvirke studiens gyldighet og generaliserbarhet. Til tross for at Tjora (2021) hevder at et mindre antall elever i fokusgruppeintervjuene kan ha bidratt til at elevene i større grad turte å dele av egne meninger og følelser, argumenterer Krueger (1994) for at en ulempe ved små fokusgruppeintervjuer er at færre perspektiver blir representert i intervjusamtalen. Dette kan bidra til at elevene fikk færre synpunkter å spille videre på i diskusjonen. På bakgrunn av det kun var 6 elever som ønsket å delta i intervju, ble jeg nødt til å gå vekk fra kravet om spredning i utvalget til fokusgruppeintervjuene. Dette kunne påvirket resultatene i form av det var en risiko for at jeg for eksempel kun fikk refleksjoner fra elever som likte undervisningsopplegget, eller som utelukkende hadde negative holdninger til det å programmere i faget. Mitt inntrykk er at elevene, særlig på spørsmålet der de ble bedt om å reflektere omkring hvorfor mange elever har negative holdninger til programmering (spørsmål 1 på intervjuguide, vedlegg H) viste at de klarte å sette seg inn i andre perspektiver enn sine egne. Dette kan ha bidratt til at flere perspektiver og synpunkter ble representert i samtalen. Ved å kunne sammenligne datamaterialet fra intervju med data fra spørreskjema fikk jeg en bekreftelse på at elevene i intervjuene i stor grad representerte holdninger og erfaringer som var gjeldende for flertallet av elevene i klassen. Det at jeg kunne bruke funn fra én metode, for å bekrefte funn fra en annen metode, er en fordelene som følger det at studien bygger på metodetriangulering.

All kvalitativ forskning er tolkning, og det er sannsynlig at andre kunne tolket resultatene på en annen måte enn det jeg hadde gjort her. For å kunne styrke gyldigheten til resultatene i denne studien kunne det ha vært gjennomført forskertriangulering. Det ble på bakgrunn av denne potensielle begrensningen valgt å inkludere tabell med kodeskjema og vedlegg som viser hvordan jeg har tolket ulike tema og sitater i fremvisningen av det kvalitative datamaterialet.

I resultatkapitlet ble det fremhevet hvordan åtte av de totalt tjue påstandene viste til signifikant effekt på elevenes holdninger. Det er viktig å anerkjenne at over halvparten av påstandene ikke viste en signifikant forskjell ved sammenligningen. I tillegg vil det ligge en begrensning til oppgavens metode, som handler om at all datainnsamling er hentet fra én enkelt biologklasse, og at sammenligningsgrunnlaget er basert på kun de 21 eleven som var til stede både ved pre- og postundersøkelsen. Det er derfor viktig å tolke resultatene i lys av disse forholdene, og det er fornuftig å stille spørsmålstegn til hvorvidt en kan stole på at resultatene fra den parvise t-testen representerer elevenes faktiske opplevelser. Jeg vil likevel påstå at det er grunn til å tro på at resultatene kan vise til tendenser og interessante perspektiver, særlig når de sees i lys av resultatene fra den kvalitative analysen (Brevik & Mathé, 2021). Flere av funnene fra den parvise t-testen, støttes av elevenes refleksjoner i fokusgruppeintervjuene og de åpne spørsmålene på spørreskjemaene. Det at flere av elevenes graderte påstander viste til en signifikant forskjell tre uker etter gjennomføringen, øker sannsynligheten for at opplegget faktisk hadde en mer *vedvarende* effekt på elevenes mer stabile holdninger til å programmere i biologi 2. Uansett viser studiens funn til interessante perspektiver og tendenser det kan være verdt å ta med seg på veien videre.

I dette tilfellet handler studiens *generaliserbarhet* om i hvilken grad jeg kan generalisere oppgavens funn til å handle om andre elever i lignende situasjoner. I hvilken grad kan funn om elevenes holdninger og effekten av et spesifikt undervisningsopplegg, også være gjeldende i andre biologiklasserom? En av de mest fremtredende svakhetene ved en studie som denne, er størrelsen på utvalget, da gjennomføring av undervisningsopplegg og datainnsamling er gjennomført i kun én enkelt klasse. Det ville vært verdifullt å ha gjennomført opplegget i flere klasser, så utvalget ble større, og slik at man i større grad kunne sagt noe om studiens funn kan sies å representere flere biologielevers holdninger og opplevelser. Oppgavens begrensende omfang understreker derfor viktigheten av å se studiens funn i lys av den konteksten og det utvalget den er gjennomført i.

Flere av egenskapene ved utvalget kan ha påvirket overføringsverdien av studiens resultater, som at elevene i den aktuelle biologiklassen ifølge faglærer er både faglig sterke og motiverte. Dette kan ha påvirket det samlede inntrykket som er brukt for å beskrive elevens holdninger og opplevelser med undervisningsopplegget. Det er for eksempel ikke sikkert studien ville gitt samme resultat dersom elevene i mindre grad var motiverte for faget. Det kan også være at studiens funn er mindre relevante for andre kontekster der elevene har mer eller mindre erfaring med programmering. På bakgrunn av at forkunnskapene til elevene i denne aktuelle klassen kan sies å representere situasjonen det i dag er å finne i mange andre klasserom (Eliassen et al., 2021; Munthe, 2022; Zukanovic, 2021), vil dette trolig

styrke studiens generaliserbarhet. Forhåpentligvis er utvalg og metode beskrevet på en slik måte at leseren kan gjøre seg opp en mening om hvorvidt studiens resultater er overførbare i andre sammenhenger. Uansett vil oppgaven utgjøre et empirisk bidrag - som i lys av tidligere studier og teori - kan si noe om biologielevers holdninger til programmering, og muligheten for å stimulere disse ved hjelp av et spesifikt undervisningsopplegg.

Kapittel 7: Avsluttende betraktninger

Oppgavens hensikt var å undersøke hvordan undervisningsopplegget «Code to Cure» kunne påvirke biologi 2 elevers holdninger til å programmere i faget. Til studien ble det spesifisert tre forskningsspørsmål, der RQ1 og RQ2 dannet bakgrunn for å kunne argumentere for RQ3, som ligger tett opp mot oppgavens hensikt. Resultatene og den påfølgende diskusjonen har vist at det er grunn til å tro at undervisningsopplegget «Code to Cure» har hatt en positiv påvirkning på elevenes holdninger til å programmere i faget. Før gjennomføringen bar elevenes holdninger preg av en negativ følelsesmessig innstilling, en lav oppfatning av egen kompetanse og flere forestillinger som stilte spørsmålstegn ved programmeringsverdi i biologi og undervisningen av biologi 2. Disse holdningene har i stor grad blitt formet på bakgrunn av elevenes tidligere erfaringer med programmering i skolen. Gjennom «Code to Cure» har elevene nå fått en *ny erfaring* med programmering i faget, som trolig og forhåpentligvis har hatt en positiv påvirkning på elevenes holdninger til å programmere – generelt og i biologi 2.

Flere implikasjoner kan følge av resultatene presentert i denne oppgaven. Resultatene fra studien har vist seg å slutte opp om tidligere forskning som fastslår viktigheten og relevansen av å forske på holdninger blant elever (Baser, 2013; Di Martino & Zan, 2010; Erol, 2020; McLeod, 1992). Resultatene stemmer godt overens med tidligere forskning som har vist flere utfordringer ved å integrere programmering inn i skolefagene, som læreres mangel på programmeringsteknisk og programmeringsdidaktisk kompetanse (Munthe, 2022; Vinnervik, 2021), og elevenes manglende forkunnskaper i programmering (Eliassen et al., 2021; Munthe, 2022; Zukanovic, 2021). Yukselturk og Altıok (2017) argumenterer i sin artikkel for at den største utfordringen handler om elevenes holdninger til verktøyet. Ved økt forståelse for hvordan de tre holdningsdimensjonene - følelsesmessig innstilling, oppfatning av egen kompetanse og syn på programmering, er avgjørende for elevenes holdninger, kan undervisere i større grad ta sikte på å forme undervisningsopplegg som stimulerer elevers holdninger til kompetansen, slik at de i større grad opplever mestring i faget (Di Martino & Zan, 2010).

Jeg vil påstå at en av de viktigste implikasjonene som følger av denne oppgaven, er en ide og et håp om at elevenes holdninger til programmering kan påvirkes. Til tross for at holdninger kjennetegnes for å være forholdsvis stabile (Hannula et al., 2004; McLeod, 1992), viser denne studien at enkeltopplevelser kan påvirkes elevenes holdninger i en positiv retning. På samme måte som i studien til Erol (2020), ble det i denne studien funnet at valg av undervisningsopplegg var avgjørende for denne endringen av elevenes holdninger. Dette spesifikke undervisningsopplegget tok utgangspunkt i anbefalinger og retningslinjer fra litteraturen, samt informasjonen som ble samlet for å belyse forskningsspørsmål 1 og 2. Gjennomføringen og elevenes tilbakemeldinger gir viktig informasjon til den videre revideringsprosessen av undervisningsopplegget. Blant disse er å tilpasse nivå og støttestrukturer slik at

undervisningsopplegget i større grad også gir utfordringer til elever med *mer* programmeringserfaring. Et annet forslag er å legge til rette for flere diskusjoner felles i klasserommet. Vedlagt (vedlegg K) ligger alle forslagene til forbedring med begrunnelser.

Til tross for oppleggets forbedringspotensial, ble en stor del av de valgene som ble tatt evaluert som styrker ved undervisningsopplegget og avgjørende for at flertallet av elevene fikk positive opplevelser under «Code to Cure». Resultatene viser derfor at et kontekstbasert undervisningsopplegg med tilfredsstillende støttestrukturer, variasjon, som tar utgangspunkt i elevenes forkunnskaper «der de er nå» og som samtidig fremhever nytteverdien av programmering i faget – kan være gode egenskaper ved et undervisningsopplegg med programmering i biologi 2. Studien støtter på denne måten tidligere litteratur som fremhever viktigheten av valg av undervisningsopplegg (Baser, 2013; Di Martino & Zan, 2010; Erol, 2020; Munthe, 2022; Sengupta et al., 2013) når målet er skape undervisning som bidrar til økt engasjement, økt mestring og en bevisstgjøring av programmeringens nytteverdi i og for faget.

Det blir viktig å ta i betraktning at elevene som deltok i denne studien har forholdsvis *få* forkunnskaper innen programmering. På nåværende tidspunkt er det fremdeles kun 2 år siden programmering kom inn i norske læreplaner som en del av fagfornyelsen. Mest sannsynlig vil det med årene som kommer skje et skifte, og elevene vil forhåpentligvis ha flere forkunnskaper og bedre forutsetninger for å bruke programmering i fagene (Munthe, 2022). Denne utviklingen gjelder trolig også for lærere, som sannsynligvis med mer erfaring vil kunne utvikle verdifulle kompetanser om hvordan undervisning med programmering bør gjennomføres for å gi elevene positive læringsopplevelser, der programmeringen bidrar for faget – og ikke omvendt.

Frem til flere av disse forkunnskapene er på plass, eksisterer det et behov for undervisningsopplegg med programmering i biologi. I faget står programmering nevnt i den overordnede delen av læreplanen, og det blir derfor mye opp til hver enkelt lærer hvordan hen velger å prioritere plass til programmering i undervisningen. Skal programmeringsaktiviteter prioriteres i løpet av skoleåret, tror jeg det blir avgjørende å få på plass engasjerende undervisningsopplegg der programmeringen virker for faget, og ikke omvendt. I tillegg til å belyse noen viktige aspekter ved elevers holdninger i møte med programmering i biologi 2, kommer denne oppgaven derfor med et annet mulig bidrag til biologiundervisning og biologididaktisk forskning. Undervisningsopplegget «Code to Cure», som ble utviklet i forbindelse med denne masteroppgaven, har blitt testet ut og forslag til revideringer vil prege den videre utviklingen av undervisningsopplegget. Håpet er at en dag vil undervisningsopplegget være tilgjengelig slik at det kan tas i bruk av flere biologilærere, og bidra til positive opplevelser i norske biologi 2 klasserom.

Forslag til videre arbeid

På bakgrunn av oppgavens begrensende omfang er det underveis i prosjektet og i framskrivningen av oppgaven, tatt flere valg om å ikke gå videre inn på enkelte elementer. Holdninger til programmering i undervisningen er et stort tema, og jeg anerkjenner at oppgavens hensikt kunne vært belyst på andre måter enn det den har blitt gjort her. Jeg anerkjenner at det er flere elementer enn de som er presentert her, som også kan være av betydning for diskusjonen om veien videre. Av andre elementer jeg gjerne skulle gått videre inn, er funnene som viste at elevene til tross for å «få til» oppgaver der de skal programmere, likevel vurderer egen kompetanse som lav i møte med programmeringsaktiviteter. Resultatet introduserer et interessant perspektiv, om at elevene trolig har behov for å forstå hva det er som skjer og hva de gjør – for å oppleve mestring. Det holder trolig ikke å få riktig svar. Det ville vært svært interessant å videre undersøke denne sammenhengen mellom elevenes oppfatning av egen kompetanse og behov for en «dypere» forståelse i egne læringsprosesser med programmering.

Ekstraoppgaven om kryssingsskjema som ble utviklet til «Code to Cure» ble noe nedprioritert i gjennomføringen og følgende også i denne oppgaven, på bakgrunn av mangel på tid. Oppgaven er ifølge faglærer svært relevant i forhold til skriftlig eksamen i emnet. Motivasjonen for å utvikle oppgaven kom derfor delvis på bakgrunn av at faglærer så et behov, i form av at flere elever ofte synes slike utregninger er tidkrevende og kompliserte, der programmering ville være svært verdifullt i biologi 2. Dersom vi på dagen for gjennomføringen hadde mer tid til disposisjon, er min antagelse at det kunne vært verdifullt dersom elevene brukte mer tid på denne oppgaven. Til tross for at det i denne studien ikke ble mulig å måle om en slik gjennomgang kunne stimulere elevenes motivasjon for å utvikle og bruke en slik kode på en potensiell eksamen, har jeg en teori om at en slik «ytre» motivasjon kan føles «nærere» og mer relevant for elevene «der de er nå». At programmering er nyttig i det biologifaglige feltet kan være motiverende for mange – særlig for elevene som ønsker å studere noe innenfor biologi. Dersom elevene opplever programmering som nyttig før en bestemt vurderingssituasjon, er min teori at dette kan virke motiverende på enda flere av elevene. Videre forskning på en slik potensiell sammenheng vil derfor være interessant for å videre forstå hvordan vi som undervisere kan legge til rette for gode undervisningssituasjoner som kombinerer fag med programmering på en god måte som motiverer elevene.

Generelt vil det vært *svært* verdifullt med videre forskning på hvordan vi som undervisere kan skape og gi elevene positive møter med programmering i fagene. Får vi som lærere kartlagt de utfordringene og mulighetene elever opplever, kan vi få verdifull informasjon om veien videre. Informasjonen kan vi bruke til å forsøke å forme undervisningsopplegg som møter elevenes forutsetninger, interesser og som vekker et engasjement og en bevissthet omkring programmeringens muligheter i faget.

Referanser

- Andersen, O. G. (2022). *Programmering si rolle i matematikkfaget - frå eit elevperspektiv* [Masteroppgave, UiT].
- Baser, M. (2013). Attitude, gender and achievement in computer programming. *Online Submission*, 14(2), 248-255.
- Blikstad-Balas, M. & Dalland, C. P. (2021). Forskningsdesign - hva må du tenke på når du skal planlegge et forskningsprosjekt? I E. Andersson-Bakken & C. P. Dalland (Red.), *Metoder i klasseromsforskning: Forskningsdesign, datainnsamling og analyse* (s. 21-43). Universitetsforlaget.
- Braun, V. & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2), 77-101.
- Brevik, L. M. & Mathé, N. E. H. (2021). Mixed methods som forskningsdesign. I E. Andersson-Bakken & C. P. Dalland (Red.), *Metoder i klasseromsforskning: Forskningsdesign, datainnsamling og analyse* (s. 47-67). Universitetsforlaget.
- Cohen, L., Manion, L. & Morrison, K. (2017). *Research methods in education*. routledge.
- Dalen, M. (2011). *Intervju som forskningsmetode - en kvalitativ tilnærming* (2. utg. utg.). Universitetsforl.
- Dalland, C. P., Bjørnstad, E. & Andersson-Bakken, E. (2021). Observasjon som metode i barnehage- og klasseromsforskning. I C. P. D. Emilia Andersson-Bakken (Red.), *Metoder i klasseromsforskning: Forskningsdesign, datainnsamling og analyse* (s. 125-152).
- David, A. A. (2021). Introducing Python programming into undergraduate biology. *The American Biology Teacher*, 83(1), 33-41.
- Di Martino, P. & Zan, R. (2010). 'Me and maths': Towards a definition of attitude grounded on students' narratives. *Journal of mathematics teacher education*, 13, 27-48.
- Du, J., Wimmer, H. & Rada, R. (2016). " Hour of Code": Can It Change Students' Attitudes Toward Programming? *Journal of Information Technology Education: Innovations in Practice*, 15, 53.
- Eagly, A. H. & Chaiken, S. (1993). *The psychology of attitudes*. Harcourt brace Jovanovich college publishers.
- Eliassen, J. E., Bøe, M. V., Nederbragt, L., Berg, M. M. & Gregers, T. F. (2021). Motivasjon for beregningsorientert biologi og sammenhengen med matematikk R2 fra videregående opplæring. *Nordic Journal of STEM Education*, 4(2), 58-70.
- Erol, O. (2020). How Do Students' Attitudes towards Programming and Self-Efficacy in Programming Change in the Robotic Programming Process? *International Journal of Progressive Education*, 16(4), 13-26.
- Forrester, C., Schwikert, S., Foster, J. & Corwin, L. (2022). Undergraduate R Programming Anxiety in Ecology: Persistent Gender Gaps and Coping Strategies. *CBE—Life Sciences Education*, 21(2), ar29.
- Frønes, T. S. & Pettersen, A. (2021). Spørreundersøkelser i utdanningsforskning. I E. Andersson-Bakken & C. P. Dalland (Red.), *Metoder i klasseromsforskning: Forskningsdesign, datainnsamling og analyse* (s. 167-205). Universitetsforlaget.
- Guzman, L. M., Pennell, M. W., Nikelski, E. & Srivastava, D. S. (2019). Successful integration of data science in undergraduate biostatistics courses using cognitive load theory. *CBE—Life Sciences Education*, 18(4), ar49.

- Hannula, M., Evans, J., Philippou, G. & Zan, R. (2004). Affect in Mathematics Education-- Exploring Theoretical Frameworks. Research Forum. *International Group for the Psychology of Mathematics Education*.
- Hannula, M. S. (2002). Attitude towards mathematics: Emotions, expectations and values. *Educational studies in Mathematics*, 49(1), 25-46.
- Haraldsrud, A. D., Sveinsson, H. A. & Løvold, H. H. (2020). *Programmering i skolen*. Universitetsforlaget
- Krueger, R. A. (1994). *Focus Groups: A Practical Guide for Applied Research* (2. utg. . utg.). Sage Publications.
- Kunnskapsdepartementet. (2017). *Overordnet del - verdier og prinsipper for grunnopplæringen*. . Fastsatt som forskrift ved kongelig resolusjon. Læreplanverket for Kunnskapsløftet 2020. . <https://www.regjeringen.no/no/dokumenter/verdier-og-prinsipper-for-grunnopplaringen/id2570003/>
- Kvale, S. & Brinkmann, S. (2015). *Det kvalitative forskningsintervju* (3. utg. . utg.). Gyldendal akademisk.
- McLeod, D. B. (1992). Research on affect in mathematics education: A reconceptualization. *Handbook of research on mathematics teaching and learning*, 1, 575-596.
- Munthe, M. (2022). *Press 'Run' to Improve Mathematical Expertise (PRIME)* [NMBU]. Ås.
- Nergård, T. (2015). Undervisningsvariabler og elevenes holdninger til naturfag. I P. v. Marion & A. Strømme (Red.), *Biologididaktikk* (2. utg. . utg., s. 65-80). Cappelen Damm Akademisk
- Nettskjema. (u.å). <https://nettskjema.no>
- NOU 2015: 8. (2015). *Fremtidens skole - Fornyelse av fag og kompetanser*. <https://www.regjeringen.no/contentassets/da148fec8c4a4ab88daa8b677a700292/nou201520150008000dddpdfs.pdf>
- Ozgur, C., Colliau, T., Rogers, G. & Hughes, Z. (2017). MatLab vs. Python vs. R. *Journal of data Science*, 15(3), 355-371.
- Pevzner, P. & Shamir, R. (2009). Computing has changed biology—biology education must catch up. *Science*, 325(5940), 541-542.
- Postholm, M. B. & Jacobsen, D. I. (2018). *Forskningsmetode for masterstudenter i lærerutdanningen*. Cappelen Damm Akademisk.
- Rahn, J., Willner, D., Deverick, J., Kemper, P. & Saha, M. (2019). Incorporating computer programming & data science into a guided inquiry-based undergraduate ecology lab. *The American Biology Teacher*, 81(9), 649-657.
- Sanne, A., Berge, O., Bungum, B., Jørgensen, E. C., Kluge, A., Kristensen, T. E., Mørken, K. M., Svorkmo, A.-G. & Voll, L. O. (2016). *Teknologi og programmering for alle - En faggjennomgang med forslag til endringer i grunnopplæringen - august 2016*. Mandat gitt av Utdanningsdirektoratet.
- Sengupta, P., Kinnebrew, J. S., Basu, S., Biswas, G. & Clark, D. (2013). Integrating computational thinking with K-12 science education using agent-based computation: A theoretical framework. *Education and Information Technologies*, 18, 351-380.
- Sevik, K. (2016). *Programmering i skolen*. Senter for IKT i utdanningen.
- Statped. (2021). *Hva er programmering?* . <https://www.statped.no/laringsressurser/teknologitema/programmering-for-barn-med-saerskilte-behov/programmering/?depth=0>

- Stephens, Z. D., Lee, S. Y., Faghri, F., Campbell, R. H., Zhai, C., Efron, M. J., Iyer, R., Schatz, M. C., Sinha, S. & Robinson, G. E. (2015). Big data: astronomical or genetical? *PLoS biology*, 13(7), e1002195.
- Svenkerud, S. W. (2021). Intervjuer i klasseromsforskning. I C. P. D. Emilia Andersson-Bakken (Red.), *Metoder i klasseromsforskning : forskningsdesign, datainnsamling og analyse* (s. 91-101). Universitetsforlaget.
- Tapia, M. & Marsh, G. E. (2002). Confirmatory Factor Analysis of the Attitudes toward Mathematics Inventory. *Paper Presented at the Annial Meeting of the Mid-Sputh Education Research Association*.
- Tjora, A. (2021). *Kvalitative forskningsmetoder i praksis* (4. utg. . utg.). Gyldendal.
- Utdanningsdirektoratet. (2020). *Læreplan i biologi (BIO01-02)*. Fastsatt som forskirpt. Læreplanverket for Kunnskapsløftet 2020.
<https://data.udir.no/kl06/v201906/laereplaner-1k20/BIO01-02.pdf?lang=nno>
- Vinnervik, P. (2021). *När lärare formar ett nytt ämnesinnehåll: intentioner, förutsättningar och utmaningar med att införa programmering i skolan* [Philosophiae doctor (PhD) thesis, Umeå universitet].
- Waters, J. B. (2020). *Programmering og dybdelæring i fysikk - en kvalitativ studie av elevers arbeid med programmering i fysikk 1* [Masteroppgave, UiO]. Oslo.
- Yukselturk, E. & Altiok, S. (2017). An investigation of the effects of programming with Scratch on the preservice IT teachers' self-efficacy perceptions and attitudes towards computer programming. *British Journal of Educational Technology*, 48(3), 789-801.
- Zukanovic, H. (2021). *Hva kjennetegner matematikklæreres holdninger rundt undervisning av programmering i teoretisk matematikk?* [Masteroppgave, NMBU].
- Øyehaug, A. B. (2019). Kjennetegn på undervisning som gir dyp forståelse. I L. O. Voll, A. B. Øyehaug & A. Holt (Red.), *Dybdelæring i naturfag* (s. 38-57). Universitetsforlaget.

Vedlegg

Vedlegg A: Godkjenning fra NSD

Vedlegg B: Code to Cure - læringssti

Vedlegg C: Code to Cure – link og utdrag fra undervisningsopplegget

Vedlegg D: Samtykkeskjema

Vedlegg E: Spørreskjema 1 (SS1)

Vedlegg F: Spørreskjema 2 (SS2)

Vedlegg G: Spørreskjema 3 (SS3)

Vedlegg H: Intervjuguide (med forslag til oppfølgingsspørsmål)

Vedlegg I: Påstandene (gruppert under holdningsdimensjoner m/p-verdi)

Vedlegg J: Utfyllende beskrivelse av tema (kvalitativ analyse)

Vedlegg K: Revidering av undervisningsopplegget – valg og begrunnelser

Vurdering av behandling av personopplysninger

Referansenummer
835156

Vurderingstype
Standard

Dato
14.11.2022

Prosjekttittel

Elevers holdninger til programmering i biologiundervisning (Masteroppgave)

Behandlingsansvarlig institusjon

Norges miljø- og biovitenskapelige universitet – NMBU / Fakultet for realfag og teknologi

Prosjektansvarlig

Morten Munthe

Student

Mathea Veidal Kvernhaugen

Prosjektperiode

01.10.2022 - 20.12.2023

Kategorier personopplysninger

Alminnelige

Lovlig grunnlag

Samtykke (Personvernforordningen art. 6 nr. 1 bokstav a)

Behandlingen av personopplysningene er lovlig så fremt den gjennomføres som oppgitt i meldeskjemaet. Det lovlige grunnlaget gjelder til 20.12.2023.

[Meldeskjema](#)

Kommentar

OM VURDERINGEN

Personverntjenester har en avtale med institusjonen du forsker eller studerer ved. Denne avtalen innebærer at vi skal gi deg råd slik at behandlingen av personopplysninger i prosjektet ditt er lovlig etter personvernregelverket.

Personverntjenester har nå vurdert den planlagte behandlingen av personopplysninger. Vår vurdering er at behandlingen er lovlig, hvis den gjennomføres slik den er beskrevet i meldeskjemaet med dialog og vedlegg.

VIKTIG INFORMASJON TIL DEG

Du må lagre, sende og sikre dataene i tråd med retningslinjene til din institusjon. Dette betyr at du må bruke leverandører for spørreskjema, skylagring, videosamtale o.l. som institusjonen din har avtale med. Vi gir generelle råd rundt dette, men det er institusjonens egne retningslinjer for informasjonssikkerhet som gjelder.

FORUTSETNINGER FOR VURDERINGEN

Vi legger til grunn at personopplysninger behandles innenfor NMBUs systemer, som oppgitt i meldeskjemaet. Dette gjelder også tilgang for biveileder ved OsloMet. Dersom det overføres personopplysninger til OsloMet, må det avklares med NMBU hvorvidt det er behov for databehandleravtale eller annen avtale som regulerer biveileders behandling av personopplysninger i prosjektet.

TYPE OPPLYSNINGER OG VARIGHET

Prosjektet vil behandle alminnelige kategorier av personopplysninger frem til 20.12.2023.

LOVLIG GRUNNLAG

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

PERSONVERNPRINSIPPER

Personverntjenester vurderer at den planlagte behandlingen av personopplysninger vil følge prinsippene i personvernforordningen om:

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

DE REGISTRERTES RETTIGHETER

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

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

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

FØLG DIN INSTITUSJONS RETNINGSLINJER

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

Ved bruk av databehandler (spørreskjemaleverandør, skylagring, videosamtale o.l.) må behandlingen oppfylle kravene til bruk av databehandler, jf. art 28 og 29. Bruk leverandører som din institusjon har avtale med.

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

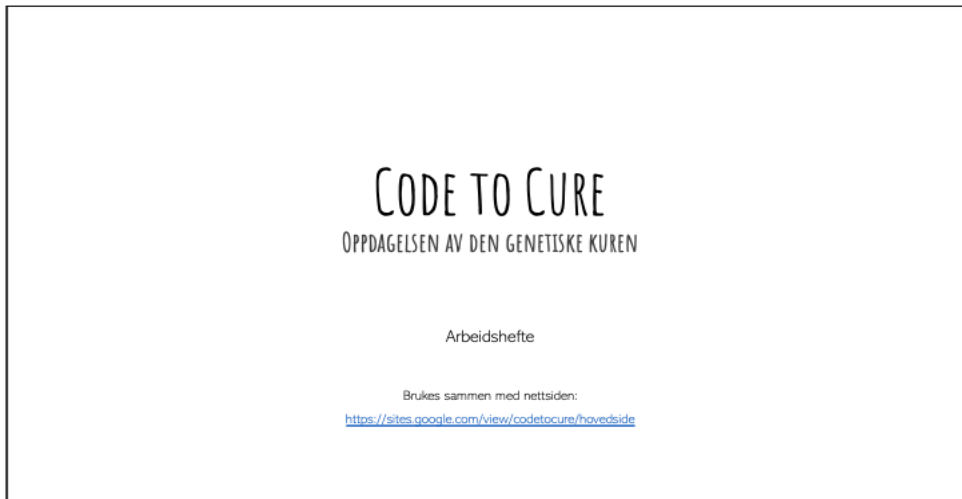
MELD VESENTLIGE ENDRINGER

Dersom det skjer vesentlige endringer i behandlingen av personopplysninger, kan det være nødvendig å melde dette til oss ved å oppdatere meldeskjemaet. Før du melder inn en endring, oppfordrer vi deg til å lese om hvilke type endringer det er nødvendig å melde: <https://www.nsd.no/personverntjenester/fylle-ut-meldeskjema-for-personopplysninger/melde-endringer-i-meldeskjema> Du må vente på svar fra oss før endringen gjennomføres.

OPPFØLGING AV PROSJEKTET

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

Lykke til med prosjektet!



1

Info før dere starter

I dette opplegget skal dere se for dere at dere er leger på et sykehus. Dere skal arbeide med følgende kompetansemål fra læreplanen:

- gjere greie for genetisk kode og regulering av genuttrykk, og beskrive korleis genteknologi kan brukast for å styre og endre genuttrykk
- gjere greie for korleis eigenskapar blir arva, og utforske årsaker til genotypisk og fenotypisk variasjon i populasjonar

Dere vil underveis i opplegget få oppgaver der dere skal programmere. Oppgavene har både hint og løsningsforslag dere kan bruke dersom dere står fast.

NB: Sykdommene dere vil bli introdusert for i opplegget er alle **fiktive** sykdommer. For at opplegget skal passe til undervisning er det underveis gjort forenklinger knyttet til diagnostikk og behandling.

2

1

Info før dere starter

Følg stegene i dette heftet. På startsiden til nettsiden ser dere de forskjellige rommene:



«Kontoret», «Venterommet» og «Laboratoriet».

Ved å følge dette heftet vil dere bli tatt mellom de forskjellige rommene.

Merk at det på «Venterommet» sitter flere pasienter. Dere skal som team kun behandle én pasient. Resten av pasientene behandles av kollegaene deres.

3

Registrere formaliteter

Velkommen til deres aller første dag på jobb!

Året er 2032. På et lite sykehus på Østlandet i Norge dere - tre ferske leger som sammen i et team skal ta i mot deres aller første pasient. Pasienten sitter på gangen utenfor og venter på å bli kalt inn på legekantoret.

Før dere slipper pasienten inn, må dere som team registrere noen formaliteter på sykehusets datasystemer:

Navn på sykehus:

Dato:

Leger på team:

Navn: _____

Navn: _____

Navn: _____

4

2

Steg 1: Diagnostikk

Tiden er inne for å slippe pasienten inn på kontoret. En viktig del av jobben som lege i møte med pasienter, er å kartlegge **symptomer**. Hvilke plager opplever pasienten som sitter i stolen foran deg?

Opplever pasienten for eksempel hoste, svimmelhet og oppkast? Ved å kartlegge symptomene kommer dere ett steg nærmere riktig diagnose.

Gå til «**Venterommet**» og finn pasienten deres. Les sykdomshistorien og forsøk å kartlegge hvilke symptomer pasienten opplever (antall symptomer vil variere)

Symptom 1: _____

Symptom 2: _____

Symptom 3: _____

Symptom 4: _____

Symptom 5: _____

5

Steg 2: Diagnostikk

- Gå til «**Kontoret**» og trykk på knappen «Sykdommen»
- Bruk funnene fra pasientmøtet og finn de 3 sykdommene som kan være aktuelle.

Mulige sykdommer:

1) _____

Defekt gen: _____ Friskt gen: _____

2) _____

Defekt gen: _____ Friskt gen: _____

3) _____

Defekt gen: _____ Friskt gen: _____

6

3

Videre diagnostikk

Livet som lege er ikke alltid like enkelt. Ofte blir man sittende å klø seg i hodet når pasientens symptomer passer overens med flere forskjellige sykdommer. Da må dere som legeteam benytte dere av flere diagnostiske metoder (undersøkelsesmetoder). Blodprøver, kliniske tester og bildediagnostikk (som røntgen, MR, CT) er eksempler på dette.

De 3 sykdommene dere mistenker er alle **genetiske sykdommer**. Dere vurderer derfor å **genteste** pasienten. Norske sykehus gjennomfører i dag genetiske undersøkelser på ca. 55 000 personer hvert år (Bioteknologirådet, 2022).

Sykehuset dere er ansatt ved er et av de fremste sykehusene i landet innen **gentesting**. Dere har derfor tilgang til et splitter nytt og avansert utstyr når dere skal diagnostisere deres pasient.

7

Steg 3: Eksperiment

Pasienten har levert en blodprøve og resultatene med pasientens DNA er nå tilgjengelig. For å finne ut av hva som feiler pasienten skal dere lete etter de genene som gir sykdom i pasientens DNA.

Dere er klare for å sette i gang, men det er ett lite problem. Gå til «**Laboratoriet**», og inn på siden som heter «**Eksperiment**».

8

4

Steg 4: Gentest

Gå til «**Laboratoriet**» og inn på siden som heter «Gentest».

Følg instruksene og finn ut av hva som feiler pasienten!

Pasienten har denne sykdommen: _____

9

Steg 5: Rapportertil overlege

Til tross for det er dere som er de behandlende legene, er det overlege på vakt som har det siste ansvaret når det kommer til diagnostikk og behandling. Alle funn og avgjørelser må derfor gå gjennom denne legen.

Gå til «**Kontoret**» og velg «Rapport».

Sjekk om overlegen er enig i resultatene dere har funnet!

10

5

Steg 6: Behandling

Nå har dere funnet ut av hva som feiler pasienten deres. Det er bra, men denne informasjonen hjelper pasienten deres lite alene.

Pasienten har en genetisk sykdom, og må få behandling slik at symptomene forsvinner.

Gå til «Laboratoriet» og velg «Genterapi».

11

Steg 7: Epikrise

Pasienten er diagnostisert og ferdig behandlet!

Det siste som gjenstår er å informere pasienten.

Gå til kontoret og velg «Epikrise»

12

Ekstra: Pasient 4

Ferdig før tiden?

Hjelp kollegaene deres OG imponer overlegen ytterligere ved å ta inn enda en pasient før helgen!

Gå til venterommet og hent inn pasient 4.

Vedlegg C: Code to Cure – link og utdrag fra undervisningsopplegget

Link: <https://sites.google.com/view/codetocure/hovedside>

Hovedsiden:

Hovedside

CODE TO CURE
OPPDAGELSEN AV DEN GENETISKE KUREN

VENTEROMMET	KONTORET	LABORATORIET
Pasient 1: Marianne	Sykdommer	Eksperiment
Pasient 2: Anton	Rapport	Gentest
Pasient 3: Georg	Epikrise	Genterapi 1
Pasient 4: Anna & Jonas		Genterapi 2

Venterommet – Pasient 1: Marianne (eksempel på sykdomshistorie):

Hovedside

Pasient 1: Marianne

Marianne er 22 år. Hun er for tiden student, og drømmer om å bli barneskolelærer. Fra tidligere kan ikke Marianne huske å ha vært noe annet enn frisk og rask. De siste månedene har dette forandret seg. Hun husker at hun for noen måneder siden la merke til at hun ofte var ekstremt sliten når hun våknet på morgenen, til tross for at hun fikk nok timer med søvn. Dette har hun ikke vært plaget med tidligere.

Hun forteller dere om episoder der hun har stått oppreist og plutselig følt seg veldig svimmel. Ved enkelte anledninger har det gått så langt at hun har besvimt. Ved lette fysiske aktiviteter, må hun nå ofte stoppe opp på grunn av problemer med pusten. Det føles ikke ut som hun får nok luft, og hun må ofte sette seg ned for å hvile.

For noen uker siden var hun i butikken og kjente at hjertet banket veldig hardt og veldig fort. Det kjentes ut som hun hadde hjertebank, og dette opplever hun nå oftere og oftere.

Flere i Mariannes omgangskrets har kommentert at hun har sett ekstra sliten ut i det siste, og hun føler hun ser blekere ut enn normalt.

Marianne begynner å bli ganske bekymret, og er lettet for at hun endelig er på sykehuset og kan få hjelp. Kan dere hjelpe Marianne med å finne ut av hva som kan være årsak til symptomene?

Kontoret – Sykdommer (eksempel på tabell der elevene skulle finne mulige sykdommer)

Her ser dere sykehusets lister over sykdommer og symptomer. Finn 3 sykdommer som passer med pasientens plager (husk å sjekke begge listene).

Under hver sykdom finner dere gensekvensene som er årsak til sykdommen (vi kaller dette "defekt" gen), samt den friske varianten av hvert gen (gensekvensen som finnes hos individer som ikke har sykdommen, vi kaller dette "friskt" gen).

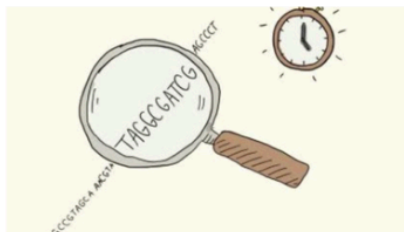
Når dere har funnet de tre mulige sykdommene, noter ned sykdommene + defekt og friskt gen til hver av disse. Dere vil trenge disse senere.

	Anemia Hematosis	Bronchia Antitrypsin	Carmichael Meningitis	Castillo Mononukleose	Erevan Syndrom	Fernandez Tuberkolose	Granville Dysentri	Greta Fever	Haruki Myodystrofi	Lungillum Mucosia
Blek hud	x					x				
Brystsmerter		x								x
Diaré					x					
Feber			x	x		x	x			
Fordøyelsesprob.									x	
Hjertebank	x				x					
Hodepine			x							
Hoste		x				x				x
Muskelsvakhet				x					x	
Hovne lymfeknuter				x						
Kvalme			x					x		
Muskelsmerter			x		x		x		x	
Pusteproblemer	x	x							x	x
Rennende nese							x			
Slim		x				x				x
Svelgeproblemer									x	
Svimelhet	x	x		x				x		x
Sår hals				x						
Utslett					x					
Vekstforstyrrelser										
Vekttap						x				
Defekt gen	CGCGCTAC	CGCTGGGT	CACTGGTC	CTTCAGAG	GGCCCCTA	GGTATTGG	TTTATCAG	ACGTGTAC	AGCCGGTC	GCATTATC
Friskt gen	CACTGGTC	TTTATCAG	GGCCCCTA	CGCGCTAC	ACGTGTAC	TAAAAG	GCAGTGGG	CGCCTCGA	GAATACACA	ACGCTGAT

Laboratoriet – Eksperimentet (oppgaven med den skeptiske legen)

Nå som dere har avgjort hvem på teamet som er den skeptiske legen, er tiden inne for å gjennomføre et lite eksperiment. Skeptikeren mener at leger med 6 års utdanning er mer rustet til å gjeneste en pasient, sammenlignet med en datamaskin. Legen mener derfor at dere bør lete gjennom pasientens DNA-sekvens manuelt.

De andre legene på teamet er uenig. Sekvensen dere finner ved å trykke på knappen nedenfor, er en DNA-sekvens som inneholder 100 000 baser.



Eksperimentet

DNA-sekvens

Den skeptiske legen skal undersøke om sekvensen GCTA er i en de to første linjene av DNA-sekvensen.

- 1) Finn frem en stoppeklokke
- 2) Mål hvor lang tid det tar å lete i to linjer av sekvensen ovenfor.
- 3) To linjer inneholder ca. 170 bokstaver (baser). Ca. hvor lang tid ville den skeptiske legen bruke på å lete gjennom hele sekvensen på 100 000 baser?

Gentest

Her skal vi forsøke å få datamaskinen til å lete etter de bestemte gensekvensene. For å få datamaskinen til å gjøre som vi vil, må vi programmere.

Se på eksempelet nedenfor. Koden viser hvordan man ved hjelp av få linjer med kode, kan få datamaskinen til å lete gjennom en sekvens med bokstaver ved hjelp av en if-setning. Avhengig av om den bestemte bokstaven er i sekvensen vi leter etter, vil programmet printe ut forskjellige beskjeder.

Eksempel 1

```
alfabet = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
bokstav = "D"

if bokstav in alfabet:
    print("Bokstaven er i alfabetet")
else:
    print("Bokstaven er ikke i alfabetet")
```

Bokstaven er i alfabetet

Bokstaven D er i variabelen som heter alfabet. Datamaskinen gir oss beskjed om at bokstaven er i alfabetet.

Eksempel 2

```
alfabet = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
bokstav = "Å"

if bokstav in alfabet:
    print("Bokstaven er i alfabetet")
else:
    print("Bokstaven er ikke i alfabetet")
```

Bokstaven er ikke i alfabetet

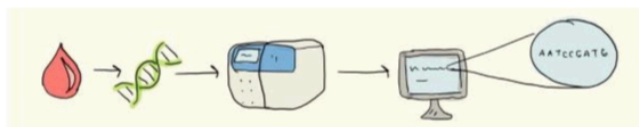
Bokstaven Å er IKKE i variabelen som heter alfabet. Datamaskinen gir oss beskjed om at bokstaven ikke er i alfabetet.

Gentest pasienten ved hjelp av programmering: ☞

Nå er det deres tur!

Ved å kjenne til pasientens symptomer, har dere eliminert en hel rekke med sykdommer. Dere sitter forhåpentligvis igjen med 3 mulige diagnoser. Hver av disse sykdommene er genetiske sykdommer, som vil si at de skyldes en "feil" i DNA-et. I dette opplegget sier vi litt forenklet at hver av sykdommene skyldes ett enkelt "defekt" gen på ca. 7 baser.

Det vil si at personen med en bestemt gensekvens, vil ha sykdommen. Har personen ikke gensekvensen, har hen heller ikke sykdommen.



Vi ønsker å utvikle et program som går i gjennom pasientens DNA og leter etter bestemte gensekvenser. Et slikt program kan vi lage på flere ulike måter, men én måte å gjøre det på er å ta utgangspunkt samme metode som i eksempelet med alfabetet ovenfor.

Gentest (fortsettelse)

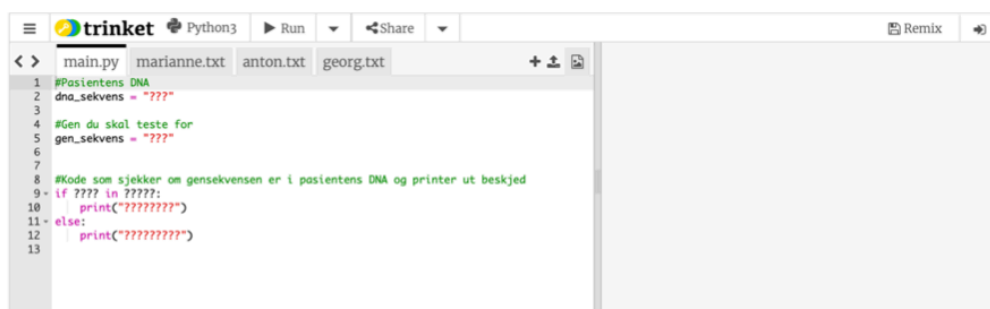
Lag programmet:

Fyll ut det som mangler nedenfor, slik at programmet printer ut en beskjed som gir brukeren informasjon om pasienten har den aktuelle gensekvensen eller ikke (obs: her er det den **defekte** gensekvensen dere skal lete etter).

Forsøk først å se om dere får det til på egenhånd. Står dere fast, kan dere etterhvert benytte dere av hintene under.

Et tips er å åpne programmerings-editoren i en ny fane (trykk på det gule nøkkelen øverst --> view on trinket.io) når dere arbeidet med programmet. Da kan dere fritt gå frem og tilbake blant sidene uten å miste arbeidet deres.

Pasientene har levert blodprøver og DNA-et har blitt isolert. Pasientens DNA finner du som en egen fane i trinket-editoren.



```
1 #Pasientens DNA
2 dna_sekvens = "???"
3
4 #Gen du skal teste for
5 gen_sekvens = "???"
6
7
8 #Kode som sjekker om gensekvensen er i pasientens DNA og printer ut beskjed
9 - if ??? in ????:
10     print("????????")
11 - else:
12     print("????????")
13
```

Hint 1

Hint 2

Variablene dere trenger er:

- 1) DNA-sekvensen til pasienten deres (ligger i egen fane i trinket-editoren)
 - 2) Sykdomsgenene (defekt gen) til de 3 forskjellige sykdommene dere vil undersøke (disse finner dere i oversikten med alle sykdommer og symptomer)
- Kopier og lim inn i koden ovenfor. Sjekk ett sykdomsgen av gangen.

[Link til løsningsforslag](#)

Ta skjermbilde av programmet deres og lagre det slik at dere enkelt finner det igjen til senere.

Fungerer programmet deres, skal dere få treff på kun én av de tre sykdommene. Noter ned navnet på denne sykdommen.

Laboratoriet- Genterapi

I denne siste oppgaven skal dere lage et program som gjør følgende:

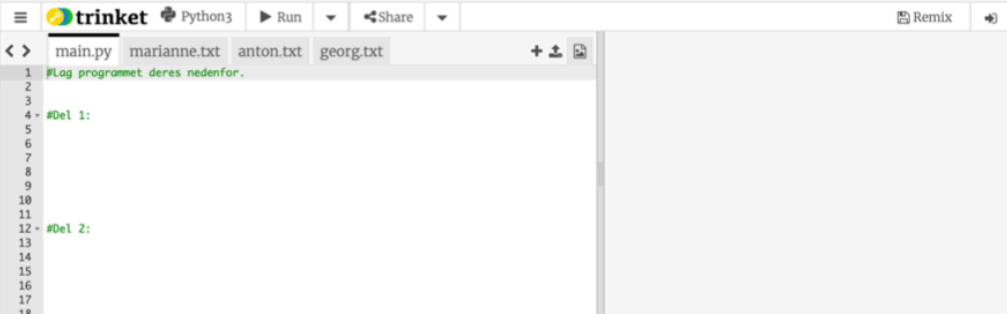
Del 1: Programmet dobbelstjekker at DNA-sekvensen faktisk inneholder sykdomsgen

Del 2: Programmet leter gjennom DNA-sekvensen, finner sykdomsgen og erstatter dette genet med et "friskt" gen

Del 3: Programmet leter gjennom DNA-sekvens og dobbelstjekker at sykdomsgenet er borte

Forsøk først på egenhånd. Står det fast, finner dere hint og løsningsforslag nedover på denne siden.

Et tips er å åpne programmerings-editoren i en ny fane (trykk på det gule nøkkelen øverst --> view on trinket.io) når dere arbeidet med programmet. Da kan dere fritt gå frem og tilbake blant sidene uten å miste arbeidet deres.



```
1 #Lag programmet deres nedenfor.
2
3
4 - #Del 1:
5
6
7
8
9
10
11
12 - #Del 2:
13
14
15
16
17
18
```

Del 1: Hint

Del 2: Hint

Del 2: Hint (replace-funksjonen)

Del 3: Hint

Løsningsforslag

Programmet fungerer og behandlingen er vellykket dersom programmet først 1) printer ut en beskjed om at pasienten har sykdommen, og så 2) printer ut en ny beskjed om at pasienten ikke lenger har sykdommen.

Ta skjermbilde av programmet deres og lagre det slik at dere enkelt finner det igjen til senere.

Ekstraoppgaven:

FLUVEUSP

Pasient 4: Anna og Jonas

Dere ble ferdig med diagnostikk og behandling av forrige pasient i rekordtid! For å imponere overlegen ytterligere OG hjelpe de andre legene på vakt (slik at de også kan ta tidlig helg) - velger dere å hente inn de siste pasientene fra venterommet. Anna og Jonas.

Anna og Jonas skiller seg fra pasienten tidligere i dag, ved at de begge føler seg helt friske. De ønsker å få barn, og kommer til sykehuset for informasjon og rådgivning.

Det har seg slik at både Anna og Jonas er bærere av gen for sykdommene cystisk fibrose og sigdcelleanemi. Cystisk fibrose er en genetisk sykdom som skyldes en mutasjon i CFTR genet, lokalisert på kromosom 7. Mutasjonen fører til forstyrrelser i kjertelfunksjonene i flere organer, og mennesker med denne sykdommen opplever mange symptomer, særlig knyttet til lungene og fordøyelseskanalen.

Sigdcelleanemi er en annen genetisk sykdom, som skyldes en mutasjon i HBB-genet, på kromosom 11. Mutasjonen påvirker produksjonen av hemoglobin, og mennesker med denne genfeilen får anemi (lav blodprosent).

Tenk dere at sykdommen cystisk fibrose blir bestemt av et gen som finnes i to alleler/genvarianter: allel C som gir fungerende CFTR-gen, og allel c som gir ikke-fungerende CFTR-gen. Se videre for dere at sykdommen sigdcelleanemi blir bestemt av et annet gen som også finnes i to alleler/genvarianter: Allelet S gir fungerende HBB-gen, og allel s gir ikke-fungerende HBB-gen.

Begge sykdommene er recessive sykdommer, som vil si at man må arve to varianter av det ikke-fungerende genet for å få sykdommen.

Anna og Jonas ønsker nå informasjon og rådgivning. For å svare på pasientenes spørsmål, skal vi lage et program i Python som hjelper oss å gjøre beregningene både raskere og mer nøyaktig. Den største delen av programmet er allerede ferdig laget, og dere skal kun fylle ut de delene av koden som mangler. Vi kaller koden dere får som utgangspunkt en "skjelettkode".

Anna og Jonas ønsker nå informasjon og rådgivning. For å svare på pasientenes spørsmål, skal vi lage et program i Python som hjelper oss å gjøre beregningene både raskere og mer nøyaktig. Den største delen av programmet er allerede ferdig laget, og dere skal kun fylle ut de delene av koden som mangler. Vi kaller koden dere får som utgangspunkt en "skjelettkode".

Før vi starter:

Her kan det lønne seg å ha to pc-skjermer å se på, slik at dere slipper å gå frem og tilbake. En av dere kan ha denne siden åpen. En annen på teamet åpner linken som fører til trinket-editoren:

<https://trinket.io/python3/9d6f340802>

PS: Ikke frykt dersom koden i editoren ser komplisert ut ved første øyekast. Vi skal gå grundig gjennom hvert steg her!

Programmets oppbygning, steg for steg:

Steg 1:



Steg 2:



Steg 3:



Samtykkeskjema

Vil du delta i forskningsprosjektet «Et forsøk for å se om valg av undervisningsopplegg endrer elevers holdninger til programmering i biologi»?

Dette er et spørsmål til deg om å delta i et forskningsprosjekt hvor formålet er å finne ut om et «virkelighetsnært» undervisningsopplegg med programmering kan endre elevenes holdninger til bruk av programmering i biologifaget. Dette informasjonsskrivet forklarer målene med prosjektet og hva en deltakelse vil innebære for deg.

Formål

I min masteroppgave skal jeg undersøke om «virkelighetsnær» undervisning kan endre elevenes holdning til bruk av programmering i biologi. Holdninger til programmering i faget handler om motivasjon, selvtilit, verdien og gleden av å bruke kompetansen i faget. Jeg ønsker å se på om et konkret undervisningsopplegg vil endre noen av disse faktorene ved å gjennomføre en spørreundersøkelse før og etter undervisningen. I tillegg ønsker jeg å velge ut enkelte elever til et fokusgruppeintervju i etterkant av opplegget.

Hvem er ansvarlig for forskningsprosjektet?

Norges miljø- og biovitenskapelige universitet (NMBU) er ansvarlig for prosjektet.

Hvorfor får du spørsmål om å delta?

I arbeidet med dette prosjektet ønsker jeg å kartlegge holdninger til elever i biologi 2. Hele klassen vil få tilbud om å fylle ut et spørreskjema. I etterkant ønsker jeg å spørre 3-5 elever som svarte på spørreskjemaet, om de ønsker å delta i et fokusgruppeintervju.

Hva innebærer det for deg å delta?

Hvis du velger å delta i prosjektet, innebærer det at du fyller ut to spørreskjema. Dette vil ta deg ca. 15 minutter. Spørreskjemaet vil inneholde påstander om dine holdninger til programmering i biologi. Spørreskjemaet inneholder påstander som «Jeg er komfortabel med å bruke programmering i biologiundervisningen», «Jeg ser hvorfor det er nyttig å kunne programmere i biologi» og «Jeg vil utvikle mine ferdigheter i programmering gjennom biologiske problemstillinger». For å kunne velge ut elever som vil få tilbud om å være med i et fokusgruppeintervju, må jeg registrere navn på spørreskjema. Skjemaet skal fylles ut to ganger, for å måle eventuelle endringer.

Dersom du blir spurt om å være med i fokusgruppeintervju vil du delta i en samtale med andre elever som omhandler holdninger til programmering i faget. Jeg som forsker vil være til stede. I tillegg vil dere få mulighet til å diskutere undervisningsopplegget som alle i klassen skal gjennomføre. Spørsmålene kan være «Hvorfor mener du det er viktig å utvikle ferdigheter i programmering» og «Hva tror du det var med undervisningsopplegg som gjorde at du endret dine holdninger til programmering i biologi?».

Både deltagere og foreldre kan få se både spørreskjema og intervjuguide i forkant dersom ønskelig.

Det er frivillig å delta

Det er frivillig å delta i prosjektet. Hvis du velger å delta, kan du når som helst trekke samtykket tilbake uten å oppgi noen grunn. Du kan trekke dette samtykket ved å ta kontakt med faglæreren din, eller sende meg (Mathea Veidal Kvernhaugen) eller veilederen min, Morten Munthe, en epost. Alle dine personopplysninger vil da bli slettet. Det vil ikke ha noen negative konsekvenser for deg dersom du ikke vil delta eller senere velger å trekke deg.

Dette gjelder både spørreskjema og intervju, og jeg som forsker har taushetsplikt.

Alle elevene i klassen skal være med på undervisningsopplegget, da dette er en del av den normale undervisningen. Spørreskjema i forkant og etterkant er frivillig, og det samme gjelder et eventuelt fokusgruppeintervju. Formålet er å finne ut hvordan min undervisning i størst mulig grad kan utvikle positive holdninger til bruk av programmering i biologi.

Ditt personvern

Vi vil bare bruke opplysningene om deg til formålene vi har fortalt om i dette skrevet. Vi behandler opplysningene konfidensielt og i samsvar med personverregelverket. Det er bare jeg og mine veiledere, Morten Munthe (NMBU) og Tonje Tomine Seland Strat (OsloMet), som vil ha tilgang til de opplysningene du gir fra deg i spørreskjema og under intervjuet.

Det eneste som vil gjenkjenne deg på spørreskjemaet ditt er navnet ditt. Spørreskjemaene vil foretas gjennom en digital kryptert løsning, med innlogging ved hjelp av FEIDE (Forms via Teams). Intervjuene blir foretatt via en app som automatisk lagrer lydfilen på en kryptert server.

I teksten vil alle navn bli anonymiserte.

Hva skjer med opplysningene dine når vi avslutter forskningsprosjektet?

Opplysningene anonymiseres når prosjektet avsluttes/oppgaven er godkjent, planlagt mai 2023. Personopplysninger, lydopptak og video vil bli destruert.

Dine rettigheter:

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

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

Hva gir oss rett til å behandle personopplysninger om deg?

Vi behandler opplysninger om deg basert på ditt samtykke.

På oppdrag fra NMBU har NSD – Norsk senter for forskningsdata AS vurdert at behandlingen av personopplysninger i dette prosjektet er i samsvar med personverregelverket.

Hvor kan jeg finne ut mer?

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

- Mathea Veidal Kvernhaugen (student)
Epost: matheaveidal@gmail.com
- NMBU ved Morten Munthe (veileder)
Epost: morten.munthe@nmbu.no
- OsloMet ved Tonje Tomine Seland Strat (biveileder)
Epost: tonjeto@oslomet.no
- Vårt personvernombud (NMBU):
Hanne Pernille Gulbrandsen
Mobil: 402 81 558
E-post: personvernombud@nmbu.no

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

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

Med vennlig hilsen,

Mathea Veidal Kvernhaugen

Samtykkeerklæring

Jeg har mottatt og forstått informasjon om prosjektet «*Elevs holdninger til programmering i biologiundervisningen*» og har fått anledning til å stille spørsmål. Jeg samtykker til:

- Å fylle ut spørreskjema i forkant og etterkant av undervisningen
- Å delta i et fokusgruppeintervju (med video/lydopptak)

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

(Signert av prosjektdeltaker, dato)



(1) Holdninger til programmering i biologi

Navn

Tusen takk for at du tar deg tid til å delta i mitt forskningsprosjekt!

I dette spørreskjemaet vil du få spørsmål om ditt forhold til biologi 2, programmering og programmering i biologifaget.

Dine svar vil forbli **anonyme**, og den eneste grunnen til at jeg ønsker at du skal oppgi navn er for at jeg senere skal kunne velge ut enkelte elever til å delta i en gruppesamtale.

Her er ingen svar mer riktig enn noen andre. Det viktigste er at du svarer på spørreskjemaet så godt du klarer, og at du svarer **så ærlig som mulig**.

Spør meg gjerne hvis det er noen av spørsmålene du lurer på!

Identifiserer du deg som:

Gutt

Jente

Annet/ønsker ikke å oppgi

Hvilket matematikkfag hadde du 1. året på vgs?

1P

1T

Annet

Hvilket matematikkfag hadde du 2. året på vgs?

2P

S1

R1

Hvilket matematikkfag har du i år?

S2

R2

Jeg har ikke matte i år

Holdninger til faget biologi 2

Hva motiverte deg til å velge biologi 2 som programfag?

Hvilket tema innenfor biologi liker du mest?

Huk av på maks 3 temaer

Evolusjon (utviklingen av arter, evolusjonsteorier, Charles Darwin, naturlig utvalg...)

Økologi (artenes levesteder og levevis, mangfold, populasjonsvekst, miljøfaktorer samspill mellom arter,...)

DNA (struktur, replikasjon, cellyklus, celledeling, proteinsyntesen...)

Genetikk (arv, nedarvingsmønstre, krysningskjema, mutasjoner...)

Bioteknologi (genteknologi, teknikker, genmodifisering, genterapi, kloning...)

Energiomsetning (aerob og anaerob celleånding, fotosyntesen...)

Hvilket tema innenfor biologi liker du minst?

Huk av på maks 3 temaer

Evolusjon (utviklingen av arter, evolusjonsteorier, Charles Darwin, naturlig utvalg...)

Økologi (arternes levesteder og levevis, mangfold, populasjonsvekst, miljøfaktorer samspill mellom arter,,,))

DNA (struktur, replikasjon, cellyklus, celledeling, proteinsyntesen...)

Genetikk (arv, nedarvingsmønstre, krysningsskjema, mutasjoner...)

Bioteknologi (genteknologi, teknikker, genmodifisering, genterapi, kloning...)

Energiomsetning (aerob og anaerob celleånding, fotosyntesen...)

Hvor godt liker du faget biologi 2?

Liker veldig godt

Liker godt

Hverken liker eller misliker

Liker lite

Liker veldig lite

Hvorfor liker du/liker du ikke biologi 2?

Hvor enig er du i påstandene?

Dere har den siste perioden hatt bioteknologi og genteknologi som tema i biologitimen.

Hvor enig er du i påstandene nedenfor?

Jeg føler meg trygg på at jeg mestrer læreplanmålet som sier at jeg "skal kunne beskrive hvordan genteknologi brukes for å styre og endre genuttrykk".

Veldig enig

Enig

Hverken enig eller uenig

Uenig

Veldig uenig

Jeg synes det er vanskelig å forstå hvordan man kan redigere gener ved hjelp av genteknologi

Veldig enig

Enig

Hverken enig eller uenig

Uenig

Veldig uenig

Jeg synes det er spennende å lære om bioteknologi og genredigering

Veldig enig

Enig

Hverken enig eller uenig

Uenig

Veldig uenig

Holdninger til programmering (generelt):

Programmerte du på ungdomsskolen?

- Ja
- Nei
- Husker ikke/vet ikke

Hvor mye programmeringserfaring har du fra videregående?

- Ingen erfaring
- Programmert litt i et/få fag
- Programmert noen ganger i flere fag
- Programmert mange ganger i flere fag

Hvis du har noe programmert tidligere - i hvilken sammenheng/fag var dette?**I hvor stor grad føler du at du mestrer oppgaver der du må programmere?**

- I svært stor grad
- I stor grad
- Nøytral
- I liten grad
- I svært liten grad

Hvor enig er du i påstandene?**Jeg liker programmering**

- Helt enig
- Enig
- Nøytral
- Uenig
- Helt uenig

Jeg synes programmering er vanskelig

- Helt enig
- Enig
- Nøytral
- Uenig
- Helt uenig

Jeg blir motivert av programmering

- Helt enig
- Enig
- Nøytral
- Uenig
- Helt uenig

Jeg opplever at jeg mestrer oppgaver med programmering

- Helt enig

Enig
Nøytral
Uenig
Helt uenig

Det er nyttig å kunne programmere

Helt enig
Enig
Nøytral
Uenig
Helt uenig

Jeg har tro på at jeg kan løse oppgaver med programmering

Helt enig
Enig
Nøytral
Uenig
Helt uenig

Jeg blir nervøs bare av tanken på å skulle programmere

Helt enig
Enig
Nøytral
Uenig
Helt uenig

Programmering er kjedelig

Helt enig
Enig
Nøytral
Uenig
Helt uenig

Jeg forstår hvorfor vi burde lære programmering på skolen

Helt enig
Enig
Nøytral
Uenig
Helt uenig

Når jeg programmerer har jeg lite selvtillit

Helt enig
Enig
Nøytral
Uenig

Helt uenig

Jeg tror jeg vil få bruk for programmering senere i livet

Helt enig

Enig

Nøytral

Uenig

Helt uenig

Jeg tror programmering kan bli nyttig i fremtiden

Helt enig

Enig

Nøytral

Uenig

Helt uenig

Når vi får oppgaver med programmering, vet jeg at jeg ikke kommer til å klare den

Helt enig

Enig

Nøytral

Uenig

Helt uenig

I samtale med en elev på en annen videregående skole, kommer det frem at eleven synes programmering er vanskelig og lite motiverende.

Hva tenker du kan være en mulig årsak til at eleven opplever programmering på denne måten?

Holdninger til programmering i biologi:

Har du noen tidligere erfaringer med programmering i biologi? Hvis ja - hvordan synes det dette var?

Hva tenker/føler du om at programmering skal inn i biologifaget?

Hvor enig er du i påstandene?

Jeg er komfortabel med å bruke programmering i biologiundervisningen

Helt enig

Enig

Nøytral

Uenig

Helt uenig

Det er spennende å programmere i biologitimene

Helt enig

Enig

Nøytral

Uenig
Helt uenig

Jeg tror programmering er nyttig hvis man vil studere eller jobbe innenfor biologi

Helt enig
Enig
Nøytral
Uenig
Helt uenig

Jeg blir mindre motivert av å måtte programmere i biologi

Helt enig
Enig
Nøytral
Uenig
Helt uenig

Det er IKKE viktig å kunne programmere i biologifaget

Helt enig
Enig
Nøytral
Uenig
Helt uenig

Jeg er motivert for å utvikle mine ferdigheter i programmering gjennom biologiske problemstillinger

Helt enig
Enig
Nøytral
Uenig
Helt uenig

Jeg kan tenke meg mange tilfeller der det kan være nyttig å programmere i faget biologi

Helt enig
Enig
Nøytral
Uenig
Helt uenig

Har du noen eksempler på når det kan være nyttig å bruke programmering i biologi?

Om undervisningsopplegg

Hjelp den fortvilte biologilæreren

På en familiemiddag kommer du i snakk med en familievenn som jobber som biologilærer på en videregående skole. Hun skal lage et undervisningsopplegg med

programmering i biologi 2, og ønsker derfor dine innspill.

Hun ber deg tenke på en time du har hatt tidligere med programmering. Hvordan synes du dette var? Hva var bra med opplegget, og hva var mindre bra?

Til slutt spør familievennen deg hva du tenker er viktig når en lærer lager undervisningsopplegg med programmering i biologi.

Hva svarer du?

Hvordan kan biologilæreren legge til rette for elevene i klassen skal få en positiv opplevelse med programmering? Kom gjerne med så mange og konkrete tips som mulig!

Tusen takk for at du tok deg tid til å svare på alle spørsmålene!! :)

(2) Holdninger til programmering i biologi

Tusen takk for at du tar deg tid til å delta i mitt forskningsprosjekt!

I dette spørreskjemaet vil du få spørsmål om ditt forhold til biologi 2, programmering og programmering i biologifaget.

Dine svar vil forbli **anonyme**, og den eneste grunnen til at jeg ønsker at du skal oppgi navn er for at jeg senere skal kunne velge ut enkelte elever til å delta i en gruppesamtale.

Her er ingen svar mer riktig enn noen andre. Det viktigste er at du svarer på spørreskjemaet så godt du klarer, og at du svarer **så ærlig som mulig**.

Spør meg gjerne hvis det er noen av spørsmålene du lurer på!

Navn

Hvor enig er du i påstandene?

Påstander om **programmering generelt**.

Jeg liker programmering

Helt uenig

Uenig

Nøytral

Enig

Helt enig

Jeg synes programmering er vanskelig

Helt uenig

Uenig

Nøytral

Enig

Helt enig

Jeg blir motivert av programmering

Helt uenig

Uenig

Nøytral

Enig

Helt enig

Jeg opplever at jeg mestrer oppgaver med programmering

Helt uenig

Uenig

Nøytral

Enig

Helt enig

Det er nyttig å kunne programmere

Helt uenig

Uenig

Nøytral
Enig
Helt enig

Jeg har tro på at jeg kan løse oppgaver med programmering

Helt uenig
Uenig
Nøytral
Enig
Helt enig

Jeg blir nervøs bare av tanken på å skulle programmere

Helt uenig
Uenig
Nøytral
Enig
Helt enig

Programmering er kjedelig

Helt uenig
Uenig
Nøytral
Enig
Helt enig

Jeg forstår hvorfor vi burde lære programmering på skolen

Helt uenig
Uenig
Nøytral
Enig
Helt enig

Når jeg programmerer har jeg lite selvtillit

Helt uenig
Uenig
Nøytral
Enig
Helt enig

Jeg tror jeg vil få bruk for programmering senere i livet

Helt uenig
Uenig
Nøytral
Enig
Helt enig

Jeg tror programmering kan bli nyttig i fremtiden

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Når vi får oppgaver med programmering, vet jeg at jeg ikke kommer til å klare den

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Hvor enig er du i påstandene?

Påstandene er knyttet til dette spesifikke undervisningsopplegget.

Jeg likte undervisningsopplegget

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Jeg forstår hvorfor programmering var en del av undervisningsopplegget

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Opplegget gjorde meg mer motivert for å programmere

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Nivået på programmeringen var på et for høyt nivå

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Programmeringen førte til at jeg lærte noe

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Jeg mistet motivasjonen når jeg jobbet med programmeringsoppgavene

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Nivået på det faglige (biologi-delen) var på et for høyt nivå

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Jeg forstår hvorfor programmering er viktig for å analysere store mengder data (feks DNA sekvenser)

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Opplegget var ikke utfordrende nok

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Det var nyttig med hint og løsningsforslag på programmeringsoppgavene

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Jeg ville likt opplegget bedre dersom vi jobbet individuelt

- Helt uenig

Uenig
Nøytral
Enig
Helt enig

Jeg ville likt opplegget bedre dersom vi gjorde mer felles (med læreren)

Helt uenig
Uenig
Nøytral
Enig
Helt enig

Det var lett å forstå hva vi skulle gjøre underveis i opplegget

Helt uenig
Uenig
Nøytral
Enig
Helt enig

I hvor stor grad føler du at du mestret oppgavene der dere skulle programmere?

I svært stor grad
I stor grad
Nøytral
I liten grad
I svært liten grad

I hvor stor grad fikk du hjelpen du trengte fra oss lærere hvis du sto fast?

I svært stor grad
I stor grad
Nøytral
I liten grad
I svært liten grad

I hvor stor grad fungerte oppsettet?

I hvor stor grad fungerte det å følge arbeidshefte samtidig som dere navigerte på nettsiden?

I svært stor grad
I stor grad
Nøytral
I liten grad
I svært liten grad

Biologi og programmering i en "virkelighetsnær oppgave"

I dette opplegget var biologien og programmeringen satt i en kontekst hvor dere som leger skulle diagnostisere og behandle en pasient. Hva tenker du om denne måten å lære på? Gjorde konteksten deg mer/mindre motivert? Forklar kort.

Til neste gang:

Deres erfaringer med opplegget er verdifulle for veien videre. Se for deg at en biologilærer spør deg om råd slik at han/hun kan forbedre opplegget til en neste gang.

Hva fungerte bra? Var det noe ved opplegget du likte?

Kom gjerne med så mange og så konkrete tilbakemeldinger som mulig.

Hva fungerte mindre bra? Hva kunne vært gjort annerledes?

Kom gjerne med så mange og så konkrete tilbakemeldinger som mulig.

Tusen takk for at du tok deg tid til å delta i undersøkelsen!

(3) Holdninger til programmering i biologi

Tusen takk for at du tar deg tid til å delta i mitt forskningsprosjekt!

Nå har det gått noen uker siden dere gjennomførte undervisningsopplegget "Code to Cure", der dere som leger skulle diagnostisere og behandle en pasient ved hjelp av programmering.

Les og avgjør hvor enig du er i påstandene nedenfor.

Navn

Hvor enig er du i påstandene?

Jeg føler meg trygg på at jeg mestrer læreplanmålet som sier at jeg "skal kunne beskrive hvordan genteknologi brukes for å styre og endre genuttrykk".

- Helt uenig
- Uenig
- Hverken enig eller uenig
- Enig
- Helt enig

Jeg synes det er vanskelig å forstå hvordan man kan redigere gener ved hjelp av genteknologi

- Helt uenig
- Uenig
- Hverken enig eller uenig
- Enig
- Helt enig

Jeg synes det er spennende å lære om bioteknologi og genredigering

- Helt uenig
- Uenig
- Hverken enig eller uenig
- Enig
- Helt enig

Hvor enig er du i påstandene?

Jeg er komfortabel med å bruke programmering i biologiundervisningen

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Det er spennende å programmere i biologitimene

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Jeg tror programmering er nyttig hvis man vil studere eller jobbe innenfor biologi

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Jeg blir mindre motivert av å måtte programmere i biologi

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Det er IKKE viktig å kunne programmere i biologifaget

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Jeg er motivert for å utvikle mine ferdigheter i programmering gjennom biologiske problemstillinger

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Jeg kan tenke meg mange tilfeller der det kan være nyttig å programmere i faget biologi

- Helt uenig
- Uenig
- Nøytral
- Enig
- Helt enig

Tusen takk for at du tok deg tid og deltok i forskningsprosjektet mitt! :)

Generert: 2023-05-10 21:21:56.

Spørsmål til diskusjon (m/ oppfølgingsspørsmål)

1. *Studier viser at programmering i skolen kan fører til at elever opplever lavt engasjement, lav motivasjon og lav mestringsfølelse.*

Hvorfor tror dere det er slik?

Hvilke tidligere erfaringer har dere med programmering?

(I hvilke situasjoner/fag)

(Hva har kjennetegnet disse oppleggene?)

(Hvordan synes dere dette har vært?)

Hva tenker dere om at programmering skal bli en del av biologifaget?)

(Hva er utfordringer/muligheter?)

2. Hva tenker dere om dette spesifikke undervisningsopplegget?

Hva skilte det fra tidligere opplegg dere har gjort med programmering?

Hva likte dere? Hvorfor?

Hva likte dere mindre? Hvorfor?

Hva var mest utfordrende?

Fikk dere den hjelpen dere trengte?

Kom det dere skulle gjøre tydelig nok frem?

Hva burde vært gjort annerledes?

Hva burde vi beholde?

3. Har opplegget på noen måte endret deres syn på programmering i biologi – i så fall hvordan?

Hva tenkte dere om programmering i biologi før?

Hvorfor tror dere opplegget har forandret deres syn?

4. Hva mener dere kjennetegner et godt undervisningsopplegg med programmering i biologi?

Hva har vært bra med tidligere undervisning?

Hva har vært mindre bra?

Har dere noen tips til lærere som skal lage opplegg med programmering?

Hva kunne gjort annerledes i dette opplegget? Hvorfor?

Vedlegg I: Påstandene (gruppert under holdningsdimensjoner m/p-verdi)

Påstand	Pre	Post 1	Post 2	P verdi (t-test)
	Før	Retten etter	3 uker etter	
Følelsesmessig innstilling (generelt)				
1. Jeg liker programmering	1,857142857	2,380952381		0,102035353
2. Jeg blir motivert av programmering	1,523809524	2,095238095		0,042285589
3. Jeg blir nervøs bare av tanken på å skulle programmere	3,38095238	3,38095238		1
4. Programmering er kjedelig	3,66666667	3,42857143		0,30855909
Følelsesmessig innstilling (biologi)				
1. Det er spennende å programmere i biologitimen	2,095238095		2,857142857	0,007725265
2. Jeg blir mindre motivert av å måtte programmere i biologi	3,904761905		3,66666667	0,469555826
3. Jeg er motivert for å utvikle mine ferdigheter i programmering gjennom biologiske problemstillinger	2,285714286		2,80952381	0,141718789
Syn på egen kompetanse (generelt)				
1. Jeg synes programmering er vanskelig	3,904761905	2,380952381		0,000521843
2. Jeg opplever at jeg mestrer oppgaver med programmering	1,80952381	2,76190476		0,00356498
3. Jeg har tro på at jeg kan løse oppgaver med programmering	2,476190476	3,047619048		0,117154358
4. Når jeg programmerer har jeg lite selvtillit	4,142857143	3,571428571		0,103565114
5. Når jeg får en oppgave i programmering jeg at jeg ikke kommer til å klare den	3,8	2,75		0,02700364
Syn på egen kompetanse (biologi)				
1. Jeg er komfortabel med å bruke programmering i biologiundervisningen	1,71428571		2,38095238	0,01550902
Syn på programmering (generelt)				
1. Det er nyttig å kunne programmere	4	3,80952381		0,492950061
2. Jeg forstår hvorfor vi burde lære programmering på skolen	3,71428571	3,80952381		0,80002814
3. Jeg tror jeg vil få bruk for programmering senere i livet	2,952380952	3,142857143		0,42759875
4. Jeg tror programmering kan bli nyttig i fremtiden	4,04761905	4,38095238		0,2596148
Syn på programmering (biolog)				
1. Jeg tror programmering er nyttig hvis man vil studere eller jobbe innenfor biologi	3,428571429		3,80952381	0,22540127
2. Det er IKKE viktig å kunne programmere i biologifaget	2,857142857		2,095238095	0,00323617
3. Jeg kan tenke meg mange tilfeller der det kan være nyttig å programmere i biologi	2,714285714		3,66666667	0,004386644
* p-verdi < 0,05 er markert i rødt				

Vedlegg J: Utfyllende beskrivelse av tema (kvalitativ analyse)

Utfyllende beskrivelse av tema (kvalitativ analyse)

RQ1 – Hva former holdninger?

Tema 1a – Faktorer som påvirker

Koden «Lærerens rolle og tilrettelegging» ble brukt på utsagn om lærerens programmeringstekniske- og programmeringsdidaktiske kompetanse, samt lærerens valg av arbeidsmetoder, støttestrukturer, tema og prioriteringer i undervisningen.

Koden «Elevenes forutsetninger» ble brukt på beskrivelser som handlet om elevenes forkunnskaper som avgjørende faktorer ved tidligere erfaringer. Utsagn som «Vi startet for sent med programmering i skolen» og «Jeg har ikke nok grunnleggende kunnskaper i programmering» ble analysert som uttrykk for elevenes forkunnskaper.

Tema 1b– Elevenes affektive opplevelser

Dette temaet beskriver elevenes «her og nå» affektive opplevelser ved tidligere programmeringsaktiviteter. Koden «følelser» inneholder elevenes utsagn om engasjement (lavt/høyt), motivasjon (lav/høy), nervøsitet/bekymring, frustrasjon, fortvilelse og en følelse av å bli overveldet. Felles er at disse beskrivelsene handler om elevenes opplevelse ved en spesifikk programmeringskontekst. Utsagn som «Det var kjedelig», «Det var ikke noe gøy» og «Ikke mange var engasjerte» ble tolket som uttrykk for lavt engasjement. Utsagn som «Jeg ga opp» og «Flere var ikke så motiverte» ble kodet som uttrykk for lav motivasjon. Affektive opplevelser preget av lavt engasjement, lav motivasjon, frustrasjon, nervøsitet eller følelse av å bli overveldet ble i analysen kodet som uttrykk for *negative affektive opplevelser*. Beskrivelser som ga uttrykk for engasjement og motivasjon ble uttrykk for *positive affektive opplevelser*.

Temaet inneholder også koden «mestringsfølelse», som inkluderer elevenes beskrivelse av opplevd mestringsfølelse ved tidligere programmeringsaktiviteter. Utsagn som «Jeg skjønnte ikke» og «Jeg fikk ikke til», ble kodet som uttrykk for lav mestringsfølelse. I tillegg ble utsagn som «Det var vanskelig» i de tilfellene der oppgavens vanskelighetsgrad førte til at elevene ikke fullførte eller forsto oppgaven, også kodet som uttrykk for lav mestringsfølelse.

RQ2 – Hva kjennetegner holdninger?

Tema 2a – Følelsesmessig innstilling (Holdningsdimensjon 1)

Temaet som representerer den første holdningsdimensjonen («Følelsesmessig innstilling») inneholder utsagn der elevene beskriver mer stabile følelser til programmering, i motsetning til følelser knyttet til enkeltopplevelser (se tema 1b). Disse følelsesmessige innstillingene ble kodet som enten positive, negative eller nøytrale. Utsagn som «Jeg liker ikke programmering», «Jeg ble bekymret da vi skulle programmere i biologi» og «Jeg har ikke lyst til å programmere i biologi» ble kodet under «negativ følelsesmessig innstilling», mens utsagn som «Programmering kan være morsomt» og «Jeg liker programmering» ble kodet som positive. Utsagn som «Jeg føler meg nøytral til programmering» og «Jeg har ikke noe forhold til det» ble kodet under «nøytral følelsesmessig innstilling».

Tema 2b – Oppfatning av egen kompetanse (Holdningsdimensjon 2)

I analysen ble elevenes beskrivelser av egen kompetanse kodet som enten lav (oppfatning av egen kompetanse som lav) eller høy (oppfatning av egen kompetanse som høy). I de tilfellene der elevene begrunnet synspunkter ved å referere til lav oppfatning av egen kompetanse ved utsagn som «Jeg synes programmering er vanskelig» og «Jeg føler ikke jeg mestrer programmering», tolket jeg dette som tilfeller der elevene oppfattet egen kompetanse som lav. Tilfeller der elevenes beskrivelser inkluderte utsagn av typen «Jeg mestrer programmering», kodet jeg som høy.

Merk at elevenes refleksjoner omkring egen mestring ved *spesifikke* enkelthendelser med programmering, ikke ble kodet som en del av «Oppfatning av egen kompetanse». Disse tilfellene ble kodet som «lav mestringsfølelse» ved tidligere erfaringer (se tema 1b). Jeg valgte på denne måten å skille mellom den mer stabile holdningsdimensjonen «Oppfatning av egen kompetanse» og opplevd mestring til én spesifikk programmeringsaktivitet. Som et eksempel kan en elev oppleve lav mestring ved én spesifikk

programmeringsoppgave i biologi, men dette trenger ikke nødvendigvis føre til at eleven oppfatter egen kompetanse innen programmering som lav. Har eleven for eksempel flere andre opplevelser der hen har opplevd mestring, trenger ikke eleven oppfatte egen kompetanse som lav. I analyseprosessen var det tilfeller der det var vanskeligere å skille mellom kodene, men jeg har etterstrebet å skille lav mestring ved enkeltstående hendelser og en mer generell oppfatning av egen kompetanse som lav.

Tema 2c – Syn på programmering (Holdningsdimensjon 3)

Holdningsdimensjonen «Syn på programmering» viste seg gjennom elevenes beskrivelser av programmering i det kvalitative datamaterialet. Opprinnelig var tanken kun å undersøke elevenes oppfatning av programmeringens *nytteverdi i faget biologi*, som underdimensjon av syn på programmering. Gjennom analysen presenterte det seg likevel flere interessante forestillinger som preger elevenes samlede syn på programmering, og som jeg derfor har valgt å inkludere disse som en del av resultatene.

Jeg har valgt å kode forestillingene så nært datamaterialet som mulig, og i resultatkapitlet vil jeg presentere de hovedforestillinger som samlet ser ut til å beskrive hoveddelen av elevenes syn på programmering. Disse underkodene er samlet i kodene «positivt syn på programmering» og «negativt syn på programmering». Et positivt syn på programmering innebærer i denne analysen de oppfatningene om programmering som verdifullt i dagens biologi, og verdifullt i den forstand at det kan bidra til økt læring i biologi 2. Et negativt syn vil i analysen omfatte forestillinger som sier det motsatte. Her blir det viktig å presisere valget om å kode utsagn som «programmering kan være nyttig i andre fag, men jeg ser ikke nytteverdien i biologi» som et negativt syn. Utsagnene er derfor vurdert opp mot elevenes forestillinger om programmering ut ifra en *biologifaglig kontekst*, og det blir viktig å forstå begrepet «negativt» ut ifra denne definisjonen.

RQ3 – Elevenes opplevelser med «Code to Cure»

Tema 3a – Generelle meninger om undervisningsopplegget

Elevenes mer generelle opplevelser og meninger om undervisningsopplegget «Code to Cure» ble samlet under dette temaet, og kodet som enten negative eller positive. De positive meningene representerer her de beskrivelsene om undervisningsopplegget som kan tolkes som opplevd motivasjon og engasjement. Utsagn som «Det var gøy», «Det var spennende» og «Jeg likte undervisningsopplegget» faller eksempelvis inn under denne koden. Utsagn som beskriver det motsatte og viser til lav motivasjon og engasjement, kodes som negative opplevelser, sammen med beskrivelser som viser til at elevene opplevde bekymring, frustrasjon og en følelse av å bli overveldet.

Tema 3b – Sentrale faktorer ved undervisningsopplegget – styrker og svakheter

Konkrete aspekter elevene trekker frem ved undervisningsopplegget faller inn under temaet *faktorer ved undervisningsopplegget*. Disse faktorene kom til uttrykk som *styrker og svakheter*, gjennom elevenes besvarelser på spørsmålene «Hva fungerte bra med undervisningsopplegget? Var det noe du likte?» og «Hva fungerte mindre bra med undervisningsopplegget? Hva kunne vært gjort annerledes?».

Tema 3c – Undervisningsoppleggets effekt på elevenes holdninger

Temaet *Effekt på elevenes holdninger* ble kodet på bakgrunn av de tre holdningsdimensjonene beskrevet i det kvalitative rammeverket. Hoveddelen av dette datamaterialet kom gjennom transkripsjonene fra fokusgruppintervjuene, ved elevenes besvarelser på «Har opplegget på noen måte endret deres syn på programmering i biologi – i så fall hvordan?». Utsagn som «Jeg ble mer motivert for å lære programmering» ble kodet som effekt på holdningsdimensjonen *følelsesmessig innstilling*. Utsagn som «Opplegget gjorde at jeg forsto programmering bedre» og «Opplegget viste meg at programmering kan være nyttig i biologi», ble kodet som effekt på henholdsvis *oppfatning på egen kompetanse og syn på programmering*.

Vedlegg K: Forslag til revidering av undervisningsopplegget – valg og begrunnelser

Forslag til revidering av «Code to Cure»

Basert på observasjoner og tilbakemeldinger fra elevene etter gjennomføringen, ville jeg gjort flere endringer på undervisningsopplegget dersom det skulle blitt gjennomført igjen.

På bakgrunn av at flere av elevene mente programmeringsoppgavene kunne **vært mer utfordrende**, ville en mulig forbedring ved opplegget være å skjule flere av de grundige forklaringene, slik at elevene først fikk mulighet til å prøve på egenhånd. Dette kunne bidratt til at elevene i større grad fikk mestringfølelse og eierskap til egen løsning, i tillegg til at elevene måtte tenke mer/utforske mer på egenhånd. Versjonen som elevene gjennomførte, hadde allerede **flere hint «gjemt»** slik at elevene aktivt måtte klikke seg inn på hintene for å få tips på veien videre. Flere av forklaringene til oppgaven kunne basert på elevenes evalueringer vært inkludert på samme måte, slik at elevene selv kunne bestemme om de ville benytte seg av støttestrukturene eller ikke. Jeg ville likevel beholdt selve programmeringsoppgavene på samme nivå, for å sikre at elevene opplever mestring underveis – som tross alt var et av de viktigste målene med undervisningsopplegget.

Andre endringer kunne vært gjort med henhold til **programmeringsplattform og hjemmesiden** elevene brukte i gjennomføringen. Til tross for flertallet av elevene trakk frem nettsiden som en styrke ved undervisningsopplegget, har plattformen Google sites også noen ulemper i undervisningssituasjoner med programmering, slik som ved «Code to Cure». ~~Trinket~~-editorene ga elevene mulighet til å gjennomføre programmeringen på samme nettside som resten av oppgavene. Likevel gjorde løsningen det vanskelig for elevene å lagre arbeidet fra tidligere oppgaver, som helt klart er en svakhet ved løsningen. Det ble her imidlertid løst ved at elevene tok skjermbilder av løsningene sine underveis, og ved at opplegget ble utviklet slik at lagring av variabler og tidligere arbeid var mindre avgjørende.

Det ville likevel vært en stor fordel om opplegget kunne vært lagt til en løsning der elevene fikk mulighet til å interagere med en nettside, slik som Google Sites, der deler av opplegget er fordelt i ulike rom, samtidig som tidligere arbeid var tilgjengelig på en bedre måte. I tidligere versjoner av opplegget, ble det lagt til ~~Lupyter~~ Notebook, der en slik lagring er mulig. Denne løsningen ga likevel mindre muligheter til å lage «rommene» elevene navigerte mellom, slik at det ble en fordeling av all informasjon og oppgaver elevene skulle gjennom. Jeg har ikke lyktes med å finne en plattform som integrerer disse funksjonalitetene på en god måte, men er klar over fordelene og ulempene ved hver av løsningene. Jeg ville derfor trolig fortsatt gjennomført opplegget slik det ble i denne studien, i påvente av en bedre løsning.

En annen endring som kunne vært gjort for å forbedre gjennomføringen ytterligere, er å sette av **mer tid til arbeidet med ekstraoppgaven**. Flere av gruppene fikk ikke tid eller valgte å ikke begynne på ekstraoppgaven når de var gjennom hoveddelen av opplegget. Ekstraoppgaven kan trolig ha stor gevinst i form av nytteverdi, for eksempel på en eksamen der elevene kan få spørsmål som inkluderer kryssningsskjema, og beregninger av utspaltingsforhold og sannsynligheten for avkom å nedarve bestemte egenskaper. Dersom elevene hadde fått oppleve verdien av programmering i denne konteksten, vil det trolig kunne øke motivasjonen for å lære verktøyet, og få frem viktigheten av programmering i faget. I en ny versjon ville jeg fremhevet denne nytteverdien enda tydeligere, antagelig ved at elevene først skulle gjøre de samme utregningene for hånd. På denne måten ville elevene forhåpentligvis oppdage både hvor effektivt programmering faktisk er, men også hvor nøyaktig den gjør arbeidet som vi «mennesker» ofte kan gjøre små feil underveis. Denne oppbygningen for å få frem verktøyets nytteverdi kunne derfor vært litt på samme måte som «den skeptiske legen» - der elevene først skulle lete etter gensekvensen manuelt, før de gjorde samme prosess bare med programmering.

Skulle jeg gjennomført opplegget én gang til, ville jeg lagt opp til at elevene i større grad skulle jobbe i **grupper på to**. Resultatene viste at samtlige av gruppene fullførte oppgavene, og flertallet av elevene rapporterte at de mestret programmeringsoppgavene. Jeg kunne redusert gruppestørrelsen for å i større grad sikre at elevene ble aktive i samarbeidet. Enkelte av gruppene jobbet i grupper på to, og det så ut til at dette fungerte like godt – hvis ikke bedre akkurat for denne oppgaven.

Opplegget kunne i større grad lagt opp til en **felles gjennomgang i klassen**. Til tross for at det kun var én enkelt elev som savner en felles gjennomgang, tror jeg flere av elevene kunne fått utbytte av en kort oppsummering eller diskusjon i etterkant av undervisningsopplegget. I en slik gjennomgang kunne elevene delt hvilke sykdommer deres pasienter hadde, og resultatene fra oppgaven med den «skeptiske legen». Videre kunne man diskutert fordeler og ulemper ved å bruke programmering i gentesting. På denne måten kunne elevene i større grad lært av hverandre, på tvers av gruppene, og utvekslet ideer og synspunkter om verktøyets betydning i faget.



Norges miljø- og biovitenskapelige universitet
Noregs miljø- og biovitenskapelige universitet
Norwegian University of Life Sciences

Postboks 5003
NO-1432 Ås
Norway