

Masteroppgave

Januar, 2022

Masteroppgave

Henrik Storesund

NTNU
Norges teknisk-naturvitenskapelige
universitet
Fakultet for naturvitenskap
Institutt for fysikk

Henrik Storesund

Beregningsorientering i fysikkutdanning

Januar 2022

Beregningsorientering i fysikkutdanning

Henrik Storesund

Fysikk og matematikk - sivilingeniør

Innlevert: Januar 2022

Hovedveileder: Trine Høyberg Andersen

Medveileder: Magnus Strøm Kahrs

Norges teknisk-naturvitenskapelige universitet
Institutt for fysikk

Beregningsorientering i fysikkutdanning

En mixed methods studie av studenters erfaringer med beregningsorientering i fysikkutdanning.

Av Henrik Storesund

TFY4900 Fysikk, masteroppgave

Veiledere: Trine Højberg Andersen og Magnus Strøm Kahrs

Januar 2022

Norges teknisk-vitenskapelige universitet

Fakultet for naturvitenskap

Institutt for fysikk



Sammendrag

Realfagsutdanningen blir i større og større grad påvirket av teknologisk utvikling for å tilpasse seg arbeidslivets behov. Moderne teknologi gir mulighet for folk flest til å gjøre enorme beregninger, og det er av stor nytte i fysikkutdanningen. Ved å utnytte datamaskiners evne til å gjøre store beregninger, kan generelle likninger omformes til numeriske uttrykk for å simulere virkelige hendelser. I den forbindelse har programmering blitt en grunnleggende del av ingeniørutdanningen. Men programmering er et omfattende fagområde som det er krevende å bli komfortabel med. For studentene kan introduksjonen av programmering fort oppleves som nok et element man skal mestre, i et utdanningsprogram som oppleves krevende nok i utgangspunktet. Dessuten er det ikke nok at studentene bare lærer seg å programmere, de må også kunne koble programmeringsferdighetene sine til fysikk og beregningsorientering.

Denne masteroppgaven inntar perspektivet til fysikkstudentene som sitter midt i prosessen med å lære seg programmering, matematikk og fysikk, alt på en gang. For å gjøre dette er det først lagd en spørreskjemaundersøkelse der en stor gruppe fysikkstudenter spørres om ulike aspekter ved deres motivasjon for å drive med beregningsorientert fysikk. Deretter intervjues et utvalg av studenter individuelt for å få en dypere forståelse for hvordan beregningsorientert fysikk oppleves for enkeltindividet. På den måten bidrar prosjektet til økt innsikt i hvordan en fysikkutdanning kan gi studentene det de trenger for å møte samfunnets krav på en måte som opprettholder studentenes motivasjon gjennom hele studiet.

Masteroppgaven baserer seg på at sentrale faktorer for motivasjon er tanker om egne ferdigheter, interesse, nytte og kostnad. Den konkluderer med at mange fysikkstudenter opplever redusert mestringsforventning i løpet av det første studieåret og at dette er knyttet til et sprik i nivå mellom ulike emner og manglende opplæring i forkant av programmeringsoppgavene. Studentenes interesse for beregningsorientering er derimot jevn og høy i de første studieårene. Det anbefales å gjøre beregningsorientering mer integrert i fysikkemner ved å inkludere opplæring i å løse beregningsorienterte oppgaver i undervisningen, gjøre de beregningsorienterte oppgavene mer relatert til resten av pensum eller ha beregningsorienterte oppgaver på eksamen.

Abstract

Science education is to a greater and greater degree affected by technological developments to adjust to the demands of working life. Modern technology grants most people the ability to perform large computations, which is of great use in the physics education. Traditional physics are based on general equations that describe relationships in nature on a general level. By taking advantage of computers' ability to perform large computations, these general equations are transformed into numerical expressions that can simulate real events. That is why programming has turned into a fundamental part of engineer educations. However, programming is a large field of study that cannot be learned over night. Physics education was already packed with material for students to deal with, it is not a trivial matter to find space for programming in that package. And it is not sufficient for students to just learn general programming, they additionally must be able to apply their programming skills to physics and computations.

This master's thesis sees through the lens of the physics students that are in the middle of the process of learning programming, mathematics and physics, all at once. In order to do so, I have made a questionnaire survey where a large group of students were asked about various aspects of their motivation for doing computational physics. Afterwards, students were interviewed in order to obtain a deeper understanding in how single individuals experience computational physics. Thus this project yields an increased insight in how physics education can give the students what they need to meet the requirements of society in a way that maintains their motivation throughout the course of the education.

The master's thesis is based on that key aspects of motivation are views about one's own competence, interest, utility and cost. It concludes that many physics students go through a decreased expectancy for success during their first year of study and that this is connected to a level gap between the courses and a lack of instructions before each programming exercise. On the other hand, the students' interest for computational physics is consistently high during the first years of studying physics. It is recommended to make computational physics more integrated in physics courses by teaching how to solve computational exercises in the lectures, connecting the computational assignments more to the rest of the curriculum or having computational assignments in the exams.

Forord

Jeg står nå på slutten av et fem og et halvt års løp på Fysikk og matematikk – sivilingeniørstudiet. Når jeg ser tilbake på årene som har gått, er det ingen tvil om at de har vært lærerike, spennende og fulle av nye opplevelser, men også krevende. Det koster å lære og det koster å bli mer selvgående. Spesielt de siste to årene med pandemi har krevd at vi studenter må være ekstra selvgående og utholdende, det har satt sitt preg på studietilværelsen min sitt punktum.

For ett år siden gikk jeg som fysikkstudent inn i fysikkdidaktikkens verden. Ved å skrive didaktisk prosjektoppgave og masteroppgave, har jeg både fått muligheten til å bli kjent med et nytt fagfelt og til å gi noe tilbake til studieprogrammet mitt. Det har vært spennende å sette seg inn i rammeverket for beregningsorientering i fysikkundervisning, blant annet siden det har gitt meg mer forståelse for hva som er fysikerens tiltenkte rolle i det moderne samfunnet. Overgangen fra tall og ligninger til tekst og ord har tidvis vært krevende, men i det store og hele passet meg godt.

Mine veiledere Magnus og Trine fortjener en stor takk, de har hjulpet meg hele veien med ukentlige møter, gode og gjennomtenkte forslag og konstruktiv tilbakemelding. Jeg vil også takke familie, venner, de jeg bor med og alle studentene som stilte opp i undersøkelsene mine.

Med det takker jeg for meg og retter blikket mot arbeidslivet og hva annet livet måtte bringe.

Trondheim, januar 2022

Henrik Storesund

Innholdsfortegnelse

Beregningsorientering i fysikkutdanning.....	i
Sammendrag	ii
Abstract	iii
Forord.....	iv
Innholdsfortegnelse	v
1 Introduksjon	1
1.1 Motivasjon.....	1
1.2 Bakgrunn for studien.....	2
1.3 Problemstilling og forskningsspørsmål.....	6
2 Litteraturgjennomgang.....	7
2.1 Læring og programmering.....	7
2.2 Expectancy-Value-modellen.....	12
3 Casebeskrivelse	13
3.1 Ferdighetsstrengen.....	13
3.2 Eksempeloppgave fra første årstrinn	17
3.3 Eksempeloppgave fra andre årstrinn	18
3.4 Eksempeloppgave fra tredje årstrinn	20
4 Metoder.....	22
4.1 Kvantitativ metode	23
4.1.1 Digitalt spørreskjema som datainnsamlingsmetode.....	23
4.1.2 Utforming av spørreskjema	24
4.1.3 Utsending av spørreskjema	28
4.1.4 Populasjon og utvalg	29
4.1.5 Analyse av spørreskjema	30
4.1.6 Validitet spørreskjemaundersøkelse.....	31
4.2 Kvalitativ metode	35
4.2.1 Kvalitative intervjuer	35
4.2.2 Utforming av intervjuguide	36
4.2.3 Utvalg av intervjudeltakere	37
4.2.4 Intervjuenes forløp.....	37
4.2.5 Validitet intervjuundersøkelse	37
4.2.6 Analyse av intervjuer	38
4.3 Metodediskusjon.....	40
4.3.1 Studiets design	40

4.3.2	Spørreskjemaundersøkelse	41
4.3.3	Intervjuundersøkelse.....	41
5	Resultater	42
5.1	Spørreundersøkelse.....	42
	Mestringsforventning.....	43
	Interesseverdi.....	44
	Nytteverdi.....	45
	Kostnad.....	46
	Forhåndskunnskaper	47
	Korrelasjon for hele utvalget.....	48
	Korrelasjon for studenter i første årstrinn	49
	Korrelasjon for studenter i andre årstrinn	50
	Korrelasjon for studenter i tredje årstrinn	51
	Påstander rettet mot hvert enkelt emne	53
	Frisvarspørsmål	54
5.2	Intervju	55
	Forhåndskunnskaper og -forventninger.....	55
	Motivasjon.....	55
	Mestringsforventning.....	55
	Interesseverdi.....	60
	Nytteverdi.....	60
	Kostnad.....	62
	Progresjon	62
	Systemet rundt studentene.....	63
	Kjennetegn på en god programmeringsoppgave.....	64
6	Diskusjon og konklusjon	65
6.1	Studenters motivasjon i ulike faser av ferdighetsstrengen.....	65
	Mestringsforventning.....	65
	Interesseverdi.....	67
	Nytteverdi.....	70
	Kostnad.....	72
6.2	Anbefalinger for videreutvikling av ferdighetsstrengen	73
6.3	Forslag for videre forskning.....	74
6.4	Konklusjon	75
7	Referanseliste	77
8	Vedlegg	80

Vedlegg A: Spørreskjema.....	80
Vedlegg B: Intervjuguide	87
Vedlegg C: Samtykkeerklæring.....	89
Vedlegg D: Meldeskjema.....	91
Vedlegg E: Temaer og koder fra intervjuene.....	96
Vedlegg F:	98
Vedlegg F1	98
Vedlegg F2	98
Vedlegg F3	99
Vedlegg F4.....	101

1 Introduksjon

1.1 Motivasjon

Datamaskiner, mobiler og andre digitale objekter tar stadig større plass i menneskers liv. Det gjelder også i fysikk, primært i den forstand at datamaskinenes enorme regnekraft gir oss muligheten til å løse matematiske problemer av dimensjoner man aldri har sett før. Den fysikken som fram til relativt nylig har dominert, er preget av eksakte, analytiske matematiske ligninger som tar for seg avgrensede og forholdsvis enkle systemer. Disse ligningene har imidlertid hatt stor nytteverdi for å kunne forstå og forutsi naturen. Noen eksempler på hva fysiske ligninger har gitt oss er evnen til å utnytte faseovergangene til vann som funksjon av temperatur og trykk for å lage kjøleskap og fryserer, elektronenes strømning gjennom metaller som funksjon av spenning og motstand til å gi oss strøm, lys og varme. Dette er basert på analytiske formuleringer for hvordan naturen henger sammen. Det er særlig i det numeriske domenet at maskinene utvider vår rekkevidde. Vi kan for eksempel beregne presise prosjektilbaner i verdensrommet for å sende raketter til andre måner og planeter, ved å modellere styrken og retningen på kreftene som virker på raketten i de ulike delene av reisen. I enhver situasjon der man har objekter som har krefter som virker på det, bidrar større datakraft til at slike situasjoner kan modelleres med stadig høyere presisjon og flere og flere faktorer kan tas i betraktning. Andre eksempler er å modellere luftstrømningene rundt et fly, som er avgjørende for å flyet skal holde seg svevende, eller vannstrømninger i hav og elver, som kan brukes for å produsere elektrisitet. Nesten uansett hvilket område av fysikken det er snakk om, spiller datamodellering en større og større rolle (OECD, 2019).

Kort sagt utgjør håndtering av datamaskiner, blant annet i form av programmering, en større og større del av fysikk (Malthe-Sørenssen et al., 2015). Siden ferdighetene som kreves av en fysiker endrer seg, skjer det også endringer i fysikkutdanningen, i form av en økt vektlegging av beregningsorientering og dermed et mer programmeringsrettet fokus. Dette krever omstilling for faglærere som må undervise i temaer de kanskje ikke har lært selv. Dette representerer ikke minst en omstilling for nye studenter, som i stor grad har forholdt seg til en analytisk tilnærming til problemløsning fra videregående skole. En programmeringsorientert omstilling innebærer endringer på flere ulike plan. For det første

må programmeringsferdigheter enten forutsettes som forkunnskap for å studere fysikk, eller så må studentene opplæres i å programmere i fysikkundervisningen. Den undervisningsansvarlige må velge hvor stor grad av fleksibilitet studentene har, for eksempel med tanke på programmeringsspråk. For det andre krever beregningsorientert fysikk at man bruker matematikk som er tilpasset numeriske metoder. Numerikk, lineær algebra og diskret matematikk er eksempler på grener innen matematikken som studentene trenger opplæring i for å mestre beregningsorientert fysikk. Fysikkstudiet ved Universitet i Oslo legger stor vekt på at studentene skal ha den nødvendige kompetanse innen programmering og matematikk før de begynner med beregningsorientert fysikk. I første semester er det stort fokus på matematikk og programmering (Malthe-Sørensen et al., 2015).

I denne studien undersøker jeg studenters erfaringer med og motivasjon for å gå gjennom denne omstillingen og å bli kjent med beregningsorientering.

1.2 Bakgrunn for studien

Programmering i fysikk, eller beregningsorientert fysikk, tar i bruk andre former for matematikk enn det som undervises i fysikk i videregående skole. Heller enn eksakte ligninger, følger datamaskiner algoritmer som bruker numeriske tilnæringsmetoder (Sørby og Angell, 2012). Programmeringens inntog i fysikk krever derfor at algoritmisk tenkning og modellering implementeres grundig i fysikkutdanningen (Odden et al., 2019). Nordby (2019) drøfter betydningen av konseptet algoritmisk tenkning, og beskriver det som å bryte ned et problem til enkle instruksjoner som så settes sammen til en algoritme. Algoritmisk tenkning er i hovedsak essensielt for å lage dataprogrammer, men kan også utgjøre et rammeverk for problemløsning i andre sammenhenger.

Et interessant og nyttig tema er hvorvidt evnen til å tenke algoritmisk gir kognitive ferdigheter som kan brukes i andre fagfelt. Dette ble undersøkt i forbindelse med forsøket på å implementere programmering i grunnskolen og videregående skole på 1980-tallet, ledet an av Seymour Papert, som beskrevet i Dolonen et al. (2019). I disse undersøkelsene var det vanskelig å påvise at evnen til å programmere og tenke algoritmisk kunne brukes i andre fagområder enn programmering. Innsatsen på 1980-tallet kalles den første bølgen av forsøk på få programmering inn i grunnskolen og videregående skole, nå befinner vi oss i den andre bølgen (Dolonen et al., 2019). Det går et viktig skille mellom programmeringen i

grunnopplæringen som så langt kun har vært blokkbasert, og programmering i høyere utdanning og arbeidslivet som er skriftbasert. Blokkbasert programmering er visuelt og enklere for barn å lære fordi det baserer seg på bruk av figurer og illustrasjoner. Skriftbasert programmering bruker derimot bare tekst. Hvert skriftbasert programmeringsspråk benytter sin egen syntaks. Å lære seg et programmeringsspråk kan sammenlignes med å lære seg et nytt skriftspråk. I denne oppgaven er det implisitt alltid skriftbasert programmering det refereres til, siden all programmering på universitetsnivå er skriftbasert, som den må være for å forberede studentene på arbeidslivet. Mye har skjedd siden 1980-tallet, digital kompetanse blir i økende grad relevant og går under sekkebetegnelsen «Kompetanser for det 21. århundre», sammen med blant annet problemløsning og samarbeidsevner (Dolonen et al., 2019).

Det finnes mange eksempler på forskning på programmering i realfagsundervisning. Nordby (2019) testet undervisningsopplegg på elever i videregående skole og fant at programmering fint kan slås sammen med matematikk og fysikk, men det er viktig at programmeringen ikke kommer for brått på. Eliassen (2020) fant at biologistudenter som hadde full fordypning i matematikk i videregående skole hadde større mestringsforventning i programmering og modellering, som tyder på at det er en sammenheng mellom matematikkferdigheter og programmeringsferdigheter. Stormo (2009) viste at nylig utdannede fysikkstudenter fikk mye bruk for beregningsorientering, men at bakgrunnen deres fra studiene ikke var tilstrekkelig.

Angående programmering i fysikk spesifikt, finnes det flere evalueringer av fysikkstudiene på NTNU. Det gjennomføres blant annet en såkalt dybdeevaluering, som er en periodisk evaluering av studieprogrammer. Det er en rapport fra januar 2021 som er laget av studenter som heter «Evaluering av lab og numerikk» (Haavind et al., 2021). Med numerikk referes det her til beregningsorienterte oppgaver, altså oppgaver som bruker programmering. Rapporten er basert på en spørreundersøkelse besvart av studenter hvor studentene er klassifisert etter hvilket år de begynte å studere, altså deres kull. I spørreundersøkelsen er det blant annet stilt spørsmål om hvorvidt studentene er fornøyd med opplæringen og hvordan de har tilegnet seg kunnskap. Rapporten består i hovedsak av grafer og forklarende tekst, tallene er ikke analysert med statistiske verktøy. Et tydelig resultat fra rapporten er at en stor andel av studentene svarer at mye av deres programmeringskompetanse i fysikkemner er selvlært underveis i studiet. Studentene er

jevnt over lite fornøyde eller nøytrale med tanke på opplæringen, men sier de har klart å lære seg mye likevel. Dette peker i samme retning som resultatene fra en kvalitativ prosjektoppgave jeg gjorde våren 2021 med fokus på fysikkstudenters erfaringer med programmering, nemlig at det eksisterer et såkalt kunnskapshull i beregningsorientert fysikk. Kunnskapshullet går ut på at studentene er nødt til å benytte kunnskap og ferdigheter som de ikke har fått opplæring i for å løse programmeringsoppgavene de får i fysikkemnene.

I mai 2021 kom det en evalueringsrapport som omhandler studieprogrammene MTFYMA (Matematikk og fysikk sivilingeniør) og BFY (Bachelor i fysikk) som er gjennomført av representanter fra ulike eksterne miljøer, som UiO og SINTEF (Dysthe et al., 2021). I denne rapporten foretas det blant annet en kvalitativ evaluering av innslaget av beregningsorientert fysikk i disse studieprogrammene, sett i lys av behovene i samfunnet. Forfatterne bak rapporten er positive til «ferdighetsstrengen» - et pedagogisk grep som forklares i kapittel 3.1, og argumenterer for å ha økt søkelys på beregningsorientering i fysikkutdanningen. Rapporten gir en liste med ferdigheter som forfatterne mener er relevant for naturvitenskapelige beregninger, derav blant annet å kunne verifisere og validere kjente algoritmer, utvikle en algoritmisk tenkning og å kunne implementere algoritmer med eksisterende programvare for å løse naturvitenskapelige problemer. Kodeutviklingspraksis, bærekraftperspektiver og etiske betraktninger er andre viktige temaer som nevnes. Rapporten konkluderer med at den ser noen mulige gap mellom undervisningsopplegget og behovene til samfunnet og anbefaler å integrere beregningsorientert fysikk i større grad enn det er nå, blant annet ved å etablere nye laboratorieoppgaver der beregningsalgoritmer inngår som verktøy.

Våren 2021 utførte jeg et forprosjekt til denne masteroppgaven. Prosjektet gikk ut på å undersøke studenters erfaringer med beregningsorientert fysikk, og jeg samlet inn data ved å gjennomføre kvalitative intervjuer med studenter. I intervjuene kom det frem en rekke ulike aspekter knyttet til hvordan det er å programmere i fysikk. Noen av disse aspektene undersøkes videre i denne masteroppgaven ved hjelp av en spørreskjemaundersøkelse, slik at jeg kan nå ut til et bredere spekter av studenter og finne ut om de nevnte problemstillingene gjelder generelt for mange studenter eller kun for noen få. Resultatene fra forprosjektet var delt inn i to kategorier: en som gikk på studentenes egen arbeidserfaring fra studiet og en som gikk ut på deres oppfatning av undervisningsopplegget.

Et tema som viste seg å være i fokus var at studentene kunne oppleve at det krevdes ferdigheter i beregningsorientert fysikk som de ikke hadde lært på forhånd, og dermed måtte lære seg selv. Siden beregningsorientert fysikk baserer seg på de tre fagområdene programmering, matematikk og fysikk, kan det oppleves som overveldende vanskelig hvis studentene mangler forkunnskap i flere av disse fagområdene. Kunnskapshullet ble spesielt beskrevet som et problem i det første studieåret, det kunne også variere mye fra emne til emne. Dette undersøkte jeg videre i denne masteroppgaven med en kvantitativ datainnsamling i form av et spørreskjema som ble sendt ut til samtlige studenter i første, andre og tredje årstrinn ved NTNU. I spørreskjemaet fikk studentene spørsmål om sine erfaringer generelt og fra hvert enkelt fysikkemne. Andre temaer fra de tidligere intervjuene som jeg inkluderte i spørreskjemaet, er hvor mye programmeringsferdigheter studentene hadde før studiestart og hvilken nytteverdi studentene har av beregningsorientert fysikk for fysikken generelt. For nåværende studenter var programmeringsemner kun valgfag i videregående skole. Man kunne derfor forvente å se en viss spredning i erfaring med skriftbasert programmering som studentene hadde med seg fra grunnopplæringen (10-årig grunnskole og 3-årig videregående skole). Dette kunne påvirke deres studieprogresjon. Fra og med høsten 2023 vil alle nye fysikkstudenter ha lært programmering i grunnopplæringen etter endringer i videregående skole (Utdanningsdirektoratet, 2021). Fremover er det derfor grunn til å forvente at studenter har en viss programmeringskompetanse og at denne er mer ensartet ved studiestart.

Denne oppgaven har fokus på erfaringene til studenter ved to studieprogrammer i fysikk ved NTNU, knyttet til en såkalt «ferdighetsstreng» innen programmering som ble implementert høsten 2019 for å styrke opplæringen av programmering i fysikk (se delkapittel 3.1). Viktige intensjoner bak ferdighetsstrengen er at programmeringsoppgavene skal være godt integrert i emnet som helhet og at det skal være en jevn overgang fra ett fysikkemne til det neste. Det er to studieprogrammer som bruker ferdighetsstrengen: MTFYMA og BFY. MTFYMA er et femårig sivilingeniørprogram som omtales som fysikk og matematikk. BFY er et treårig bachelorprogram som omtales som bachelor i fysikk. Ettersom ferdighetsstrengen ble implementert høsten 2019, er det kun studentene i første, andre og tredje årstrinn som har fulgt den fra begynnelsen av studieløpet.

Formålet med denne masteroppgaven er å undersøke erfaringene til studentene i hvert ledd av ferdighetsstrengen. Sentrale temaer i den sammenheng er hvorvidt studentene er motivert for å programmere i fysikk, om de opplever mestringsfølelse i forbindelse med programmeringsoppgavene, om de synes programmering er interessant å drive med og om de ser nytte i å programmere i fysikk. I den sammenhengen har jeg valgt å bruke Expectancy-Value-modellen (EV-modellen) som teoretisk modell (Eccles og Wigfield, 2002). Den deler de underliggende faktorene for menneskers motivasjon og grunnlag for å ta avgjørelser inn i de to kategoriene mestringsforventning og subjektiv verdi. Med andre ord forsøker modellen å besvare følgende spørsmål: «Hvor sannsynlig er det at jeg lykkes med dette?» og «Hvor mye får jeg igjen for å lykkes med dette?» I denne oppgavens kontekst dreier det seg om programmeringsoppgaver i fysikkemner. Jeg vil undersøke i hvilken grad studentene opplever mestring med programmering, om de synes programmering i fysikk er interessant, om de ser på programmering som noe de har bruk for og om de synes programmering i fysikk er krevende.

1.3 Problemstilling og forskningsspørsmål

Masteroppgavens problemstilling er:

Hva kjennetegner første-, andre- og tredjeårsstudenters erfaringer med beregningsorientering i fysikkundervisningen?

Innenfor denne problemstillingen har jeg valgt to forskningsspørsmål hvor erfaringer utdypes i henhold til EV-modellen og det legges vekt på utvikling i læringsprosessen gjennom studieårene:

Hva kjennetegner første-, andre- og tredjeårsstudenters mestringsforventning, interesseverdi, nytteverdi og kostnad knyttet til beregningsorientert fysikk?

Hvordan utvikler studenters erfaringer med beregningsorientert fysikk seg i løpet av studieårene som omfattes av ferdighetsstrengmodellen?

Problemstillingen omfatter et overordnet spørsmål om studenters erfaringer med beregningsorientering i fysikk. Studien vil gi et bilde av hvordan fysikkstudenten opplever opplæringen i ulike stadier av de tre første årene av studieprogrammet. Problemstillingen retter seg mot de tre første årene spesifikt siden det er disse som inngår i

ferdighetsstrengen. Ferdighetsstrengen ble dessuten implementert høsten 2019, dermed er det kun studentene i første, andre og tredje årstrinn som har opplevd strengen helt fra de gikk i første årstrinn.

Denne studien bidrar til feltet fysikkdidaktikk ved å fokusere på studentenes opplevde kompetanse innen beregningsorientert fysikk, som strekker seg ut over tre studieår og minst seks enkeltemner. Intensjonen er også at denne studien skal bidra til en informert videreutvikling av ferdighetsstrengen innen programmering for fagmiljøet ved NTNU. Siden de ansvarlige for ferdighetsstrengen har uttrykt klare ideer om punkter som strengen er ment å forbedre, er det nyttig å undersøke hva studentene synes om disse punktene. Eksempler på slike punkter er at programmeringsoppgavene skal være godt integrert i emnet som helhet og at det skal være en jevn overgang fra ett fysikkemne til det neste.

Det første forskningsspørsmålet tar for seg hver av de utvalgte dimensjonene i EV-modellen. Dette er valgt for å kartlegge hvert av aspektene ved fysikkstudentenes erfaringer i detalj og dermed skape et godt grunnlag for å avgjøre hvilke deler av ferdighetsstrengen som fungerer godt og hvilke deler som kan bli forbedret. Intensjonen er også at man ved hjelp av EV-modellen også skal kunne undersøke kunnskapshullet som ble avdekket i prosjektoppgaven, siden manglende kunnskap og kompetanse kan påvirke blant annet studentenes mestringsforventning og kostnad innen beregningsorientert fysikk.

2 Litteraturgjennomgang

2.1 Læring og programmering

Siden problemstillingen i denne oppgaven retter seg mot studenters erfaringer med utdanningen de er i ferd med å gå gjennom, er det av verdi å beskrive hva det innebærer å befinne seg i en læringssituasjon. Brown et al. (1989) gjør et skille mellom to ulike tilnærminger til læring: læring som situert, altså et resultat av omgivelsene og kulturen den befinner seg i, eller ved å gå i lære som student ved en læringsinstitusjon og å ignorere konteksten kunnskapen springer ut fra. Skillet mellom de to tilnærmingene ligger i at den første gir en direkte tilknytning mellom hvordan den nye kunnskapen brukes i samfunnet, mens den andre tilnærmingen gjør at man lærer uavhengig av noen kontekst eller kultur. Problemene som løses innen den første tilnærmingen er sett i direkte kontekst med problemenes opphav, mens problemene i tilnærming to sees uavhengig av hvordan

problemet eller behovet for å løse det oppstår. Til tross for dette, vil studenter allikevel til enhver tid forsøke å plassere ny kunnskap i en kontekst og tillegge det en mening eller nytte. Brown et al. (1989) kritiserer synet på læring som uavhengig av kontekst, selv om det typisk er måten fysikkundervisning foregår på; studentene får presentert generaliserte, kontekstfrie teorier som de deretter selv skal kunne bruke til å løse konkrete problemer. I denne sammenhengen kan beregningsorientering bidra til å løse problemstillingen Brown et al. (1989) viser til ved å gi studentene en arbeidsmetode som er mer autentisk og tettere knyttet opp til yrkesutøvelsen som fysiker.

Mye av forskningen som er gjort på undervisning og læring i programmering retter seg mot grunnopplæringen under overskriften digital tenkning (Dolonen et al., 2019). Digital tenkning er også sentralt i høyere utdanning, men på en annen måte ettersom studentene allerede skal besitte en generell forståelse av hva digital tenkning innebærer. Dette er godt illustrert ved hjelp av Dolonen et al. (2019) sin inndeling av digital tenkning i fire kategorier:

1. Digital tenkning som forståelse av verden rundt oss, som er i ferd med å bli stadig mer digitalisert.
2. Digital tenkning som programmeringskompetanse. En pragmatisk tilnærming rettet mot hvordan kunnskap om programmeringsrelatert syntaks og semantikk kan skape verdi for samfunnet.
3. Digital tenkning som algoritmisk tenkning, et verktøy innen problemløsning.
4. Digital tenkning som brede digitale ferdigheter, en kompetanse i å velge og å tilpasse hensiktsmessige digitale verktøy for bestemte formål.

Mens punkt 1 er av mer teoretisk og filosofisk natur, retter punkt 2, 3 og 4 seg mot bruksnyttene av programmering, hvilket gjør dem mer relevant i en realfagskontekst.

Weintrop et al. (2016) har utviklet en taksonomi som er utformet spesifikt med hensyn til digital tenkning i naturfagsdelen av grunnopplæringen. Den deler inn programmeringsferdigheter i fire hovedkategorier: datahåndtering, modellering og simulering, digital problemløsning og systemtenkning. Datahåndtering dreier seg om datainnsamling, databehandling og datapresentering. Modellering og simulering handler om utforming, vurdering og bruk av digitale modeller. Digital problemløsning går ut på å programmere, å velge effektive digitale verktøy og å vurdere ulike løsninger på et problem.

Systemtenkning handler om å undersøke, forstå, definere og kommunisere informasjon om systemer. De fire hovedkategoriene kan sees på som en konkretisering av punkt 2, 3 og 4 i Dolonen et al. (2019) sin modell.

En tilnærming til begrepet digital tenkning som er mer problemløsningsrettet er utformet i Shute et al. (2017), der det er fem punkter:

- Omformulering av problemet til noe kjent og løsbart
- Rekursjon for å konstruere et system inkrementelt basert på forhåndsinformasjon
- Dekomposisjon av problemet i håndterbare deler
- Abstraksjon for å modellere kjerneaspektene ved et system
- Systematisk testing for finne løsninger, feilsøking

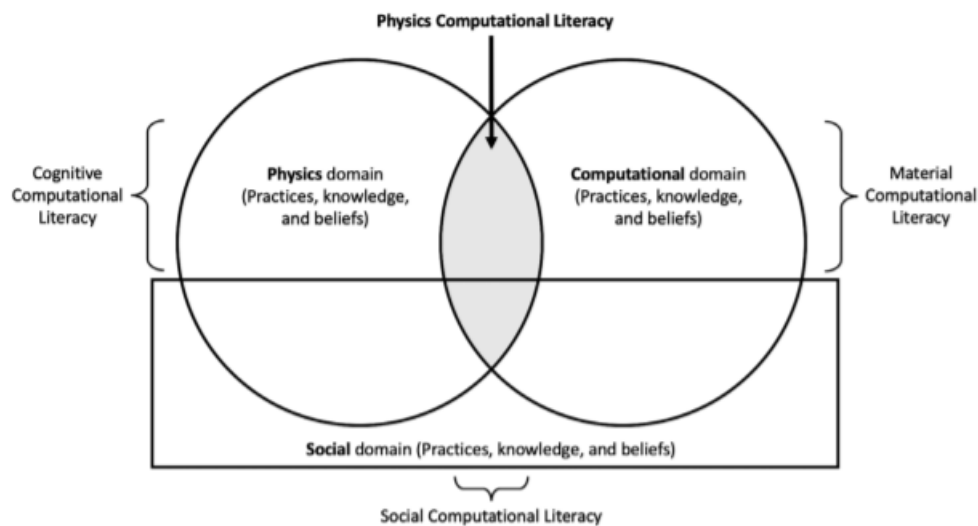
Disse punktene oppsummerer godt prosessen i å løse en oppgave i beregningsorientert fysikk, der man starter med å tenke algoritmisk for å gjøre en omfattende problemstilling mer håndterbar og deretter går gjennom punktene flere ganger for å til slutt ende opp med et velfungerende program.

Implementeringen av digitale ferdigheter i grunnopplæringen kommer ikke uten utfordringer, spesielt siden programmering er kjent for å være vanskelig å lære seg (Bennedsen og Caspersen, 2007). Innen realfag kommer programmering gjerne i tillegg til det tradisjonelle, analytiske innholdet, som gjør fagene mer komplekse og øker faren for såkalt kognitiv overbelastning (Nordby, 2019). Til tross for dette, finner ikke Bennedsen og Caspersen (2007) noen tegn til at det er spesielt høy strykprosent i programmeringsemner på universitetsnivå. Det kan peke i retning av at det er kombinasjonen mellom programmering og realfag som er kognitivt krevende.

I denne oppgaven er det programmering i fysikk spesifikt som er i søkelyset. Odden et al. (2019) utformet en tredeling av ferdighetene som inngår i beregningsorientert fysikk (figur 1):

1. Programmeringsferdigheter (material computational literacy). Alt som knyttes til det rent programmeringstekniske, som omhandler blant annet programmeringssyntaks- og semantikk, algoritmisk tenkning og generell kunnskap om datamaskiner og digitale verktøy.

2. Kognitive ferdigheter (cognitive computational literacy). Evnen til å knytte det som skjer i dataprogrammet til den fysiske virkeligheten det er ment å representere.
3. Sosiale ferdigheter (social domain). Evnen til å kommunisere om og med programmering med andre mennesker. Det kan sammenfattes i to deler: samarbeid om programmering og å lage lesbar kode.



Figur 1: Oversikt over en tredeling av ferdighetene som inngår i beregningsorientert fysikk (Odden et al., 2019).

Sammenlignet med generell programmering og beregningsorientering, er det de kognitive ferdighetene som skiller seg ut i beregningsorientert fysikk, og som kan forårsake den kognitive overbelastningen som Nordby (2019) beskriver. Fysikk er et fagfelt som i seg selv krever gode kognitive evner, siden det baserer seg på matematiske formuleringer. Ved å også inkludere programmering, er det nødvendig for å studentene å besitte et visst nivå i matematikk, fysikk og programmering for å kunne drive med beregningsorientert fysikk. En kompakt og detaljert beskrivelse av hva en moderne fysiker burde være i stand til er «å bruke ens kunnskap om teorier, metoder, modeller og implementering til å vurdere, visualisere og utforske et fysisk system ved hjelp av eksperimenter og simuleringer» (Sørby og Angell, 2012, s. 284).

Sørby og Angell (2012) demonstrerer hvor utfordrende det kognitive aspektet ved beregningsorientert fysikk kan være. Det er ikke nok å bare kunne sin matematikk, fysikk og programmering i beregningsorientert fysikk, man må også evne å koble dem sammen og bruke hvert enkelt fagfelt utenfor dets egen kontekst. Hvis ikke kan det oppstå et behov for

å operere med bestemte arbeidsmoduser som man veksler mellom og unnlater å se i sammenheng med hverandre. Studentene i undersøkelsen til Sørby og Angell ble observert å ha en «programmeringsmodus», en «matematikkmodus» og en «fysikkmodus». Programmeringsmodusen er demonstrert ved at studentene prøvde å forstå dataprogrammet gjennom kodesyntaksen heller enn i de matematiske og fysiske modellene. Matematikkmodusen går ut på å ignorere de fysiske dimensjonene variablene representerer og heller kun vurdere dem ut ifra sine matematiske egenskaper. Fysikkmodusen dreier seg om å gjøre antakelser om dataprogrammet basert på ens erfaringer om hvordan den fysiske naturen oppfører seg. Alle de tre modusene er nødvendige å kunne bruke i beregningsorientert fysikk, men det oppstår et problem når man til enhver tid kun benytter en eller to av dem og unnlater å veksle mellom dem. Sørby og Angell (2012) påpeker at de feilaktige konklusjonene som utspringer i de ulike arbeidsmodusene kan kobles til at studentene bruker helt nye verktøy som gjør det vanskelig for dem å forstå styrkene og svakhetene ved modellen som blir brukt.

Malthe-Sørenssen et al. (2015) argumenterer for at en moderne fysikkutdanning må gi like god kompetanse innen det de kaller beregninger, altså matematikk med digitale hjelpemidler, som innen tradisjonell matematikk. Et høyt nivå innen beregningsorientert fysikk innebærer å kunne slå sammen ferdigheter fra matematikk, fysikk og programmering. For at studentene skal nå et høyt nivå må de anvende beregningsorientering i fysikk regelmessig gjennom hele studiet. Et økt fokus på beregningsorientering i fysikkutdanning kan gi flere typer gevinster; studentene får bedre trening i problemløsning og økt motivasjon gjennom å få ferdigheter som er mer relevant i arbeidslivet (Malthe-Sørenssen et al., 2015).

For å forstå hva som kan gjøres for å optimalisere studenters erfaring med beregningsorientert fysikk, kan det være nyttig å vite mer om suksessfaktorer i programmeringsundervisning generelt. Wilson og Shrock (2001) analyserte kvantitativt 12 slike suksessfaktorer, derav blant annet tidligere programmeringserfaring, oppmuntring, komfortnivå og matematikkbakgrunn. Komfortnivå ble målt ved å stille studentene spørsmål om blant annet å stille og å svare på spørsmål i forelesninger og i lab, stressnivå når de gjorde programmeringsoppgaver og oppfattet vanskelighetsgrad av faget. Blant disse 12 faktorene var de to største suksessprediktorene komfortnivå og matematikkbakgrunn. Siden det krever et visst matematikkgrunnlag for å drive med fysikk, gir det grunn til å tro at

beregningsorientering i utgangspunktet egner seg godt for fysikkstudenter. Den store betydningen av komfortnivå for mestring kan tyde på flere ulike ting, blant annet at det er en sterk sammenheng mellom mestringsforventning og komfortnivå blant deltakerne. Det peker også mot at det er av stor betydning at studentene føler seg komfortable og ikke skremt i læringssituasjonen, og at de kan både stille og svare på spørsmål i og utenfor klasserommet (Wilson og Shrock, 2001).

2.2 Expectancy-Value-modellen

For å kartlegge studenters erfaringer med programmering i fysikkemner, bruker jeg Expectancy-Value-modellen (EV-modellen) (Eccles og Wigfield, 2002; Wigfield et al., 2009). EV-modellen egner seg godt til å forstå menneskers motivasjon og avgjørelsesgrunnlag. Den går ut på at motivasjon kan sees på som et produkt av to faktorer: mestringsforventning og subjektiv verdi. Med mestringsforventning menes i hvilken grad en person tror han eller hun vil få det til. Med subjektiv verdi menes hvor høyt en person verdsetter et tema. Ordet subjektiv brukes siden det samme temaet kan ha ulik verdi fra en person til en annen.

Det kan være ulike grunner til at en person har høyere mestringsforventning knyttet til et tema enn en annen, men begrepet har en klar betydning likevel. Subjektiv verdi er derimot mer sammensatt, det er en rekke ulike faktorer som tillegger et tema subjektiv verdi. EV-modellen har delt subjektiv verdi inn i fire hovedkomponenter (Eccles og Wigfield, 2002; Wigfield et al., 2009):

1. Oppnåelsesverdi – viktigheten av å lykkes med en gitt oppgave, knyttet til ens egen identitet.
2. Interesseverdi – gleden en opplever ved å drive med temaet.
3. Nytteverdi – i hvilken grad temaet passer med ens fremtidsplaner.
4. Kostnad – hvor slitsomt eller krevende det er å drive med temaet. Dette er en negativ variabel.

I min studie måler jeg variablene mestringsforventning, interesseverdi, nytteverdi og kostnad, det vil si alle komponentene som inngår i EV-modellen bortsett fra oppnåelsesverdi. Noen grunner til at oppnåelsesverdi ikke er inkludert er at jeg innenfor rammene av dette prosjektet så meg nødt til å velge bort en dimensjon, og at de andre dimensjonene dekker godt det jeg søker å finne svar på.

Bungum et al. (2012) viser at nysgjerrighet, interesse og gode resultater er noen av de viktigste motivasjonsfaktorene for fysikkstudenter. Innenfor rammene av EV-modellen betyr dette at mestringsforventning og interesseverdi har mye å si. Fremtidige yrkesmuligheter regnes derimot som mindre viktig, som tilsier at studentene ikke er så opptatt av studiets fremtidige nytte. Disse resultatene dreier seg om studentenes overordnede erfaringer med fysikkstudiet, som gir et godt bakgrunnsbilde for analysene i dette prosjektet.

3 Casebeskrivelse

Dette kapitlet gir en detaljert beskrivelse av de relevante delene av studieprogrammene til studentene som inngår i undersøkelsene. Delkapittel 3.1 gir generell informasjon om ferdighetsstrengen. Delkapittel 3.2, 3.3 og 3.4 viser til eksempeløvinger fra henholdsvis første, andre og tredje årstrinn. Delkapittel 3.1 og 3.2 er hentet fra prosjektoppgaven som ble gjort i forkant av denne oppgaven, siden øvingene dreier seg om samme case. Prosjektoppgaven ble skrevet våren 2021, semesteret før denne oppgaven. Det er gjort enkelte endringer i delkapitlene fra prosjektoppgaven for at teksten skal passe inn i strukturen i resten av kapitlet. Delkapittel 3.2, 3.3 og 3.4 benytter øvinger som er hentet fra emnesidene på Blackboard, NTNUs digitale læringsplattform.

3.1 Ferdighetsstrengen

Som nevnt i introduksjonen, ligger det en ferdighetsstrengmodell innen beregningsorientering til grunn for fysikkemnene fra første til tredje årstrinn på fysikk og matematikk sivilingeniør og bachelor i fysikk ved NTNU. I forkant av prosjektoppgaven intervjuet jeg en av de ansvarlige for ferdighetsstrengen, mye av informasjonen i dette delkapitlet er hentet fra dette intervjuet. Til forskjell fra tidligere, da programmeringsoppgavene i stor grad var formet av hver enkelt faglærer, er ferdighetsstrengen en modell som har til hensikt å skape en jevn progresjon innen programmering og beregningsorientering på tvers av fysikkemner. Formålet med ferdighetsstrengen er å sørge for at studentene ved utgangen av tredje årstrinn har ferdigheter innen beregningsorientering som er tilfredsstillende for fremtidige masteremner og arbeidsliv.

Studieprogrammene som undersøkes har de offisielle navnene MTFYMA (Matematikk og fysikk – sivilingeniør) og BFY (Bachelor i fysikk). Det inngår en rekke fysikkemner som er felles

for begge studieprogrammene. Hvert av disse emnene har en emnekode og et formelt navn, men i praksis bruker studentene gjerne forkortede versjoner. For eksempel forkortes Mekanisk fysikk til mekfys og Elektrisitet og magnetisme til elmag. Jeg vil i oppgaven holde meg til de offisielle emnenavnene, bortsett fra i kommentarer til studentsitater der forkortede versjoner er brukt.

Som en konsekvens av ferdighetsstrengen har programmeringsopplegget i de ulike fysikkemnene felles struktur: i hvert emne inngår tre programmeringsøvinger og deretter et avsluttende programmeringsprosjekt. Programmeringsoppgavene, inkludert prosjekt, teller 10 % på sluttkarakteren, men er ikke nødvendige for å kunne gå opp til avsluttende eksamen. Det inngår ikke programmeringsoppgaver på selve eksamen. Det henger sammen med at det ikke er hensikten at studentene skal lære seg å programmere i fysikkemner, det skal de ha lært i de rene programmeringsemnene.

Tabell 1 viser en oversikt over alle emnene i første og andre årstrinn på fysikk og matematikk sivilingeniør – studiet (NTNU, 2021) samt de av tredjeårsemnene som inngår i ferdighetsstrengen. Tredjeårsemnene er en kombinasjon av valgbare og obligatoriske emner. Studentene på bachelor i fysikk har de samme fysikkemnene.

Tabell 1: Oversikt over emner i første, andre og tredje år på fysikk og matematikk sivilingeniør. Fargekoder: programmering - oransje, fysikk - grønn, matematikk - mørkeblå, annet - lyseblå.

År	Semester	Emne
1	Høst	Examen philosophicum for naturvitenskap og teknologi
		Mekanisk fysikk
		Informasjonsteknologi, grunnkurs
		Matematikk 1
	Vår	Elektrisitet og magnetisme
		Bølgefysikk og fluidmekanikk
Matematikk 2		
2	Høst	Termisk fysikk
		Innføring i kvantefysikk
		Matematikk 4K
		Statistikk
	Vår	Prosedyre- og objektorientert programmering
		Optikk
		Introduksjon til vitenskapelige beregninger
		Kjemi
3	Høst	Måleteknikk

		Kjerne- og strålingsfysikk Statistisk fysikk Kvantemekanikk 1 Klassisk mekanikk
	Vår	Instrumentering Cellebiologi og cellulær biofysikk Molekylær biofysikk Faste stoffers fysikk Elektromagnetisk teori

Studieplanen består av en sammensetning av fysikkemner, matematikkemner, programmeringsemner og enkelte andre emner. De to første studieårene er det tre fysikkemner, ett programmeringsemne og tre matematikkemner per studieår. Informasjonsteknologi, grunnkurs (ITGK) i første semester gir basisferdigheter i programmeringspråket Python. I fysikkøvingene skal studentene bruke Python i et program som heter Jupyter Notebook, som er en del av en større pakke med programmeringsverktøy som heter Anaconda. Studentene har også et programmeringsemne i fjerde semester, Prosedyre- og objektorientert programmering, hvor de lærer å bruke programmeringspråket C++. Matematikkemnene er relevante innen beregningsorientert fysikk i og med at de inneholder sentrale fagfelter som lineær algebra og numerikk. Allerede i første semester i Matematikk 1 begynner studentene å lære om numeriske metoder. Lineær algebra blir studentene introdusert for i Matematikk 3 i andre semester. Oppgavene i beregningsorientert fysikk kalles gjerne også numeriske øvinger siden de i stor grad benytter numeriske tilnæringsmetoder. Studentene lærer gradvis om numerikk i et samspill mellom fysikk- og matematikkemnene. Det første rene numerikkemnet er i fjerde semester; Introduksjon til vitenskapelige beregninger (NTNU, 2021), som er obligatorisk på MTFYMA og valgbart på BFY. Siden ferdighetsstrengen strekker seg fra første til tredje årstrinn, inngår også fysikkemnene i tredje årstrinn i strengen, det til tross for at ikke alle emnene er obligatoriske for alle studenter.

Den overordnede progresjonen for ferdighetsstrengen er blant annet basert på at oppgaveformen blir friere og friere jo lenger ut i studiet man kommer. De første semestrene er oppgavene forholdsvis styrte og detaljerte med tanke på hvilke programmeringsverktøy som skal brukes. I andre og tredje årstrinn er det derimot opp til studentene hvordan de velger å løse oppgavene, kravene dreier seg i hovedsak om hvilke resultater programmet gir.

En annen gradvis overgang er at øvingene blir mer og mer omfattende. I fjerde semester kommer de første numeriske prosjektene som studentene arbeider med jevnlig hele semesteret. Fram til fjerde semester er øvingene forholdsvis kompakte og ment å skulle løses i løpet av en uke.

Det er hensiktsmessig å se ferdighetsstrengen i lys av Weintrops taksonomi (Weintrop et al., 2016). Ferdighetsstrengen tar primært for seg de delen av taksonomien som omhandler modellering, simulering og digital problemløsning, som for eksempel å velge tilnærminger til et problem, bruke digitale modeller til å finne og teste løsninger og å bruke digitale modeller for å forstå begreper. Lenger ut i studieprogresjonen legges det også noe vekt på å velge effektive beregningsverktøy, siden studentene står mer fritt til å velge programmeringsverktøy selv. Helhetlig systemtenkning er det lite av, oppgavene er som regel spesifikke, uten uttrykt tilknytning til hverandre. Men oppgavene gir god innsikt i deres begrensede systemer. Datahåndtering, den siste hovedkategorien i taksonomien, foregår kun i den eksperimentelle delen av fysikkemnene, som vanligvis ikke inkluderer beregningsorientering.

For programmeringsøvingene er det satt av egne øvingstimer med obligatorisk oppmøte der det er universitetsansatte studentassistenter (studenter som allerede har fullført emnet) til stede for å hjelpe studentene. Studentene jobber i par og det er ingen krav om å levere inn noe av arbeidet. Hensikten med å ha obligatorisk oppmøte og frivillig innlevering er å legge fokus på å arbeide med å løse oppgavene, uavhengig av om man får rett svar eller ikke, som er et viktig virkemiddel for å unngå at studentene kopierer hverandres besvarelser.

Prosjektet på slutten av semesteret er det meningen at studentene skal løse uten hjelp fra studentassistenter, og ferdighetene som kreves i prosjektet baserer seg på det studentene skal ha lært i øvingene.

3.2 Eksempeloppgave fra første årstrinn

Første eksempeleppløp (figur 2) er hentet fra en øving i emnet TFY4163 - Bølgefysikk og fluidmekanikk våren 2021, som går i andre semester. Oppgaven benytter Eulers metode, som er den samme numeriske metoden som den som ble benyttet i prosjektet i Mekanisk fysikk fra første semester. Differensialligninger er også et tema som er gjennomgått i Matematikk 1.

Oppgave 2 – Numerisk løsning av bevegelsesligningen for pendelen

a) Skriv bevegelsesligningen med tilnærmelsen $\sin \theta = \theta$ for pendelen som to første ordens differensialligninger. Diskretiser ligningen.

SKRIV HER (eller på papir)

b) Skriv en funksjon som returnerer beregnet utslagsvinkel, θ , og vinkelfrekvens, ω , opp til $t = T$ ved bruk av **Euler metoden**. Funksjonen skal ta inn tidssteget Δt , sluttiden T , samt initialverdiene for vinkelen og vinkelfrekvensen. Kommenter koden.

Om du trenger å repetere Euler metoden kan du gå gjennom Noteboken [Euler's method](#).

I kolonnen under er det gitt en del kommentarer på hva du må huske å ha med i koden, og forslag til en funksjonsdeklarasjon. Om du ønsker kan du benytte dette for å få hjelp til å starte.

Hint: Benytt `np.zeros`, til å lage en array av lengde `n+1` fylt med tallet 0, for θ og ω . Benytt `np.linspace` for å lage et array for tiden med jevnt fordelte verdier fra en startverdi til en sluttverdi.

```
In [ ]: # As you already imported numpy and matplotlib above, you do not have to import again.
# (as long as you ran the cell)
```

```
def euler_method(theta_0, w_0, dt, T):
    """
    Calculates angular displacement and angular velocity using the Euler-Cromer method.

    theta_0: initial angular displacement
    w_0: initial angular velocity (omega)
    dt: timestep
    T: max T
    """
    # SKRIV KODE HER

    """
    theta: array with values of angular displacement
    w: array with values of angular velocity
    t: array with time-values
    """
    return theta, w, t
```

c) Plot utslagsvinkelen, θ , som funksjon av tid, t , sammen med den analytiske løsningen av bevegelsesligningen (opp til $t = 10$ s). Bruk tidsstegene $\Delta t = 0.001$ s og $\Delta t = 0.004$ s og $\Delta t = 0.007$ s.

Vær nøye med navn på akser, legg til legend, og ha med tittel. Du kan øke grensene for y-aksen med `plt.ylim(min, max)`, om du ønsker det.

```
In [ ]: # SKRIV KODE HER
```

Figur 2: Eksempeloppgave fra andre studiesemester i emnet TFY4163 - Bølgefysikk og fluidmekanikk.

I oppgaven i figur 2 er det først en informerende tittel etterfulgt av deloppgaver. Deloppgave a) er som en tradisjonell, analytisk fysikkoppgave som løses for hånd på papir. Det fungerer som en introduksjon til den følgende programmeringsoppgaven: oppgave b). I oppgaveteksten i oppgave b) er arbeidskrav og informerende tekst bakt sammen, blant annet med en lenke til et hjelpedokument. Oppgaveteksten følges av et hint som inneholder en nyttig innebygd funksjon. Deretter kommer vinduet der studentene skal fylle inn kode selv, men det ligger allerede kode med forklarende kommentarer klart, slik at det skal være lettere å komme i gang. Det står i oppgaveteksten at det er valgfritt om man vil bruke den eksisterende koden eller ikke.

3.3 Eksempeloppgave fra andre årstrinn

Andre eksempeloppgave (figur 3) er hentet fra den første av to programmeringsøvinger i andreårsemnet TFY4215 - Innføring i kvantefysikk høsten 2021. Jeg har valgt å inkludere den innledende teoriseksjonen i utdraget for å gi et inntrykk av rammeverket studentene typisk forholder seg til i beregningsorientert fysikk. Temaet for oppgaven, som er gitt fra oppgavens tittel, er Schrödingerligningen, som også er et sentralt tema i regneøvingene og kvantefysikk generelt.

Numerisk løsning av den tidsuavhengige Schrödingerligningen

Å løse den tidsuavhengige Schrödingerligningen

$$\hat{H}\psi = -\frac{\hbar^2}{2m}\psi'' + V\psi = E\psi,$$

dvs. å bestemme energieigenverdier E og tilhørende energiefunksjoner $\psi(x)$ for et gitt potensial $V(x)$, er et sentralt problem i kvantemekanikken. Dette er ofte ingen enkel oppgave. Selv for potensialer som gir ligningen analytiske løsninger, kreves det ofte betydelig innsats og bruk av spesielle teknikker for å komme fram til disse. Vi skal her se på en elegant og generell teknikk for å løse ligningen numerisk for et vilkårlig potensial (i én dimensjon).

Numeriske løsningsmetoder innebærer alltid en viss avgrensning og diskretisering for å gjøre problemet endelig og håndterlig for en datamaskin. Vi avgrensner her delen av rommet vi ser på til å ligge mellom to endepunkter x_0 og x_{N+1} og deler opp intervallet mellom dem i punktene $x_0, x_1, \dots, x_N, x_{N+1}$ med lik avstand Δx mellom hvert punkt. Utenfor dette området definerer vi potensialet til $V(x \leq x_0) = V(x \geq x_{N+1}) = \infty$, slik at $\psi(x \leq x_0) = \psi(x \geq x_{N+1}) = 0$ og det kun er bølgefunksjonens verdier på rutenettet $\mathbf{x} = [x_1, \dots, x_N]^T$ som er ukjente og av interesse. Til hvert punkt tilordner vi verdiene $\psi_i = \psi(x_i)$ og $V_i = V(x_i)$ til energiefunksjonene og potensialet, og vi refererer til verdiene av funksjonene i alle punktene ved hjelp av vektorene $\mathbf{V} = [V_1, \dots, V_N]^T$ og $\boldsymbol{\psi} = [\psi_1, \dots, \psi_N]^T$.

En intuitiv og enkel tilnærming av den deriverte til en funksjon er den sentrale differansen

$$\psi'(x) = \frac{\psi(x + \Delta x/2) - \psi(x - \Delta x/2)}{\Delta x}$$

Om vi bruker denne tilnærmingen to ganger, kan vi også tilnærme den andrederiverte som

$$\psi''(x) = \frac{\psi'(x + \Delta x/2) - \psi'(x - \Delta x/2)}{\Delta x} = \frac{\psi(x + \Delta x) - 2\psi(x) + \psi(x - \Delta x)}{\Delta x^2}$$

Ved å sette denne tilnærmingen inn i den tidsuavhengige Schrödingerligningen, kan vi tilnærme den numerisk som

$$-\frac{\hbar^2}{2m} \frac{\psi_{i+1} - 2\psi_i + \psi_{i-1}}{\Delta x^2} + V_i \psi_i = E \psi_i \quad \text{for } i = 1, 2, \dots, N$$

Vi kan uttrykke denne ligningen på en elegant måte ved å innføre $N \times N$ -Hamiltonmatrisen H med elementer

$$H_{ij} = \begin{cases} \hbar^2/(m\Delta x^2) + V_i & \text{for } i = j \quad (\text{på diagonalen}) \\ -\hbar^2/(2m\Delta x^2) & \text{for } i = j \pm 1 \quad (\text{på semi-diagonalene}) \\ 0 & \text{ellers} \end{cases}$$

og benytte oss av vektoren $\boldsymbol{\psi} = [\psi_1, \dots, \psi_N]^T$. Den tar da formen

$$H\boldsymbol{\psi} = E\boldsymbol{\psi}$$

Energierne E og energiefunksjonene $\boldsymbol{\psi}$ er dermed egenverdier og egenvektorer til matrisen H !

Numerikkbiblioteker har funksjonalitet for å finne egenverdier og egenvektorer til vilkårlige matriser. De har gjerne også spesialiserte funksjoner som gjør dette mer effektivt for matriser med en spesiell form, for eksempel som den *tridiagonale* (samt reelle og symmetriske) formen til matrisen H .

Skriv en funksjon som beregner og returnerer alle energieigenverdier E og tilhørende energiefunksjoner ψ for en partikkel med masse m som befinner seg i et gitt potensial V på rutenettet \mathbf{x} . Normér energiefunksjonene i forstanden $\int |\psi|^2 dx = 1$.

I resten av øvingen skal vi rett og slett bare bruke denne numeriske løsningsmetoden på en rekke forskjellige potensialer. I noen av eksemplene skal vi også sammenligne de numeriske verdiene med analytiske resultater. For å gjøre denne prosessen så enkel som mulig, foreslår vi at du her skriver én "ultimat" plottefunksjon som du kan gjenbruke i alle disse oppgavene.

Skriv en funksjon som framstiller potensialet V , energieigenverdier E og energiefunksjoner ψ (eller absoluttkvadratene $|\psi|^2$) på rutenettet \mathbf{x} grafisk. Funksjonen skal også kunne brukes til å sammenligne to sett med (numeriske og analytiske) energier og energiefunksjoner.

Gjør gjerne dette parallelt med resten av oppgavene, slik at du kan tilpasse framstillingen basert på behovene som oppstår. Se gjerne i forelesningsnotater, bøker og søk rundt på Internett for å få litt inspirasjon til hvordan framstillingen kan gjøres.

Partikkel i boks

Et av de første kvantemekaniske problemene vi støter på er partikkel i boks. Her er potensialet, de normerte energiefunksjonene og energieigenverdiene

$$V(x) = \begin{cases} 0 & \text{for } 0 \leq x \leq L \\ \infty & \text{ellers} \end{cases}, \quad \psi(x) = \sqrt{\frac{2}{L}} \sin \frac{n\pi x}{L}, \quad E = \frac{n^2 \pi^2 \hbar^2}{2mL^2}, \quad n = 1, 2, \dots,$$

Sammenlign numeriske og analytiske verdier for noen energier og energiefunksjoner for et elektron i en boks grafisk.

Hvordan er spredningen i energinivåene?

Figur 3: Eksempeloppgave fra tredje studiesemester i TFY4215 - Innføring i kvantefysikk

Hensikten med den teoretiske introduksjonen er å omforme det presise, analytiske uttrykket som er gitt fra teorien i emnet, til en numerisk tilnærming som egner seg til beregningsorientering. Vi ser at det i utdraget er tre deloppgaver. Den konkrete oppgaveteksten til hver deloppgave er markert med fet skrift, og det er et tomt vindu klart rett under hvor studentene skal skrive inn koden.

Deloppgavene bygger på hverandre. Det gjør at man kan nytte av det man har lagd i tidligere oppgaver, som bidrar til å styrke oppgavens relevans. På en annen side medfører det at studenter som ikke får til første oppgave, vil ha problemer med å kunne gå videre til påfølgende oppgaver.

Til forskjell fra første eksempeloppgave (figur 2), er det ikke gitt noen kode i svarfeltet. Det er heller ingen konkrete råd i teksten om hvordan studentene kan strukturere koden sin, som gjør at det stilles høyere krav til studentene med tanke på programmeringsferdigheter. Oppgavene gir en viss pekepinn på kodens struktur ved å spesifisere at man skal skrive en funksjon. Utover det er det helt opp til studentene å omforme de matematiske uttrykkene i teoriseksjonen til kodestrukturer i et programmeringsspråk.

3.4 Eksempeloppgave fra tredje årstrinn

Det tredje og siste eksempelet jeg vil trekke fram, er deler av en øving fra tredjeårsemnet TFY4345 – Klassisk mekanikk fra høsten 2021. I og med at øvingen inneholder en teoriseksjon fra side 1 til side 5, har jeg valgt å begrense utdraget (figur 4) til første side og den delen av femte side som inneholder oppgaveteksten til oppgave 1. Det er verdt å nevne at ikke alle studentene som følger ferdighetsstrengen, har dette emnet. På MTFYMA er emnet kun obligatorisk for studentene som har valgt å fordype seg videre i fysikk ved å gå retningen Teknisk fysikk, som er en av tre retninger studentene kan velge mellom. For resterende studenter på MTFYMA og alle studenter på BFY er TFY4345 – Klassisk mekanikk et valgbart emne.

TFY 4345 Computational Assignment 1 : Projectile Motion

Introduction

This is the first compulsory computational exercise for the subject TFY4345 - Classical Mechanics, which has to be returned by October 11th at 13:00 on Blackboard. You may work either alone or in pairs (latter recommended). Refer to your partner when handing in the assignment. Everyone must return a report due to bookkeeping purposes (the same report for pairs, this is subject to changes if we find a more elegant solution in BB). **For crucial background information, please read the sections preceding the tasks.** Prepare your report using either LaTeX or by writing it directly into Jupyter Notebook using Markdown cells. Pay special attention on the quality of figures as these are the most important content (are the labels visible, is the line thickness appropriate, etc.). Attached source code should be zipped. You may program in any language as long as you prepare your own code from scratch. The code should be easy to read with comments and consistent naming schemes. Assessment: Accepted (1) or requires a revision (0).

Numerical calculation of a projectile trajectory

Let us consider two spatial dimensions, ignoring air resistance the equations of motion (EOM) read

$$\begin{aligned} \frac{d^2x}{dt^2} &= 0, \\ \frac{d^2y}{dt^2} &= -g, \end{aligned} \quad (1)$$

where g is the gravitational acceleration. These are second-order differential equations, to solve them numerically we will express them as a set of ordinary

1

Task 1: Projectile motion with air resistance

a) First make a program that can calculate **projectile motion in 2D without any drag**. Use the Runge-Kutta algorithm (4th order) when solving the ODE's. Note that solving Eq. (4) iteratively is the basis of the Euler method, not Runge-Kutta. Adapt your algorithm correspondingly. Compare your numerical solution with the analytical solution (derive it) and remember to consider the effect of the time step Δt used. Do a test to validate that the numerical approach converges to the analytical one when Δt is systematically decreased. Use a firing angle of $\theta = 45^\circ$ and a firing velocity of $v_0 = 700\text{m/s}$ for the initial conditions as a benchmark. Select an appropriate time step for later use.

B_2/m	$4 \times 10^{-5} \text{ m}^{-1}$
$k_B T / \bar{m} g$	10^4 m
a	$6.5 \times 10^{-3} \text{ K/m}$
α	2.5

Table 1: Constants and parameters.

b) Next, calculate the trajectory of a cannon shell **including both air drag and the reduced air density at high altitudes**. Perform your calculation for different firing angles (θ) and determine the optimal value of the angle that gives the maximum range with the following air drag models:

- Without any air density corrections, see Eq. (10).
- With the isothermic density model, see Eqs. (12), (13).
- With the adiabatic density model, see Eqs. (12), (14).

Use the parameters and constants presented in table 1.

5

Figur 4: Eksempeloppgave fra femte studiesemester i emnet TFY4345 - Klassisk mekanikk.

Øvingen i figur 4 har form som et prosjekt, som innebærer at studentene i tillegg til å programmere skal sammenfatte resultatene i en rapport. I introduksjonen til øvingen spesifiseres en rekke kriterier. Det følger en omfattende teoriseksjon, og deretter oppgavetekst på side 5. Teoriseksjonen fungerer på samme måte som i oppgaven vist i figur 3; den tar utgangspunkt i analytiske ligninger og utleder numeriske uttrykk som er relevante for øvingen. Det forventes at studentene behersker engelsk fagspråk godt, som er vanlig når studentene nærmer seg masternivå.

I motsetning til figur 2 og 3 som studentene kan åpne direkte i Jupyter Notebook, er denne øvingen forelagt i pdf-format uten tilknytting til noe bestemt dataprogram. Dette stiller høyere krav til selvstendighet. Oppgaveteksten beskriver kun de fysiske og matematiske

aspektene ved oppgaven, den gir ingen hint om hva slags kodestruktur man burde bruke eller på hva slags format svaret burde foreligge. Heller enn tydelige, oppstykkede oppgaver, består oppgaveteksten av en rekke instruksjoner som kan implementeres på mange ulike måter. Det er gitt forslag til initialbetingelser, det vil si tallverdier for relevante matematiske konstanter. Flere av konstantene har enheter, som studentene selv må avgjøre hvordan de vil presentere i koden.

Jeg har så langt i oppgaven beskrevet litteratur og bakgrunn og undersøkelsene, jeg vil gå videre med å beskrive hvordan jeg gikk fram for å gjennomføre datainnsamlinger og analyser.

4 Metoder

I dette masterprosjektet hvor jeg skal undersøke studenters erfaringer med beregningsorientert fysikk, har jeg valgt å gjennomføre først en kvantitativ tverrsnittsundersøkelse (Postholm og Jacobsen, 2018, s. 80-81) og deretter en kvalitativ intervjuundersøkelse. I litteraturen omtales denne tilnærmingen med bruk av flere metoder som mixed methods (Robson og McCartan, 2017, s. 30). Jeg bruker en spesiell variant av mixed methods som kalles et sekvensielt, forklarende design. Den går ut på å starte med en kvantitativ undersøkelse som deretter følges opp av en kvalitativ undersøkelse som søker å forklare resultatene fra den kvantitative undersøkelsen (Robson og McCartan, 2017, s. 178). Den kvantitative undersøkelsen søker å beskrive en stor populasjon gjennom tall og statistiske sammenhenger, mens den kvalitative undersøkelsen søker å beskrive fenomenet ved at et lite utvalg av populasjonen forteller ved hjelp av egne ord. I prosjektoppgaven fra våren 2021 gjennomførte jeg kvalitative intervjuer med en åpen, induktiv tilnærming for å få et inntrykk av hva studenter tenkte og hvordan de erfarte undervisningen i beregningsorientert fysikk. Basert på analysen av disse intervjuene utviklet jeg temaer. Det er med utgangspunkt i disse temaene samt eksisterende teori at jeg i denne masteroppgaven først har gjennomført en kvantitativ analyse i med utgangspunkt i et spørreskjema. Spørreskjemaet skal bidra til å svare ut problemstillingen ved å gi et generelt, empirisk bilde av studenters erfaringer med beregningsorientert fysikk, der spørsmålene i spørreskjemaet er utformet basert på teorien i litteraturgjennomgangen og temaene fra prosjektoppgaven. Ved hjelp av et spørreskjema strukturert etter EV-modellen, er hensikten

å få kunnskap om en større gruppe studenters erfaringer. I etterkant av analysen spørreskjemaundersøkelsen gjennomførte jeg kvalitative intervjuer for en utdyping av foreløpige resultater fra den kvantitative analysen. Den kvalitative delen både ut ifra en både induktivt og deduktiv tilnærming. Dette vil et bedre grunnlag for å kunne vurdere svarene fra spørreskjemaet (Postholm og Jacobsen, 2018, s. 101-104).

4.1 Kvantitativ metode

Kvantitativ metode baserer seg på å måle hyppigheten av utvalgte fenomener.

Informasjonen er standardisert; den tvinges inn i forhåndsdefinerte kategorier (Postholm og Jacobsen, 2018, s. 166). I praksis innebærer det å definere bestemte begreper som kan standardiseres ved hjelp av tall. Siden ulike mennesker i større eller mindre grad kan ha ulike assosiasjoner knyttet til de samme begrepene, kan manglende forståelse av dette fra forskeren sin side føre til resultater med lav grad av validitet (Postholm og Jacobsen, 2018, s. 166-170). Det er to måter å unngå dette problemet på:

1. Kun bruke konkrete begreper som er svært tydelige i den gjeldende konteksten og har en godt definert betydning.
2. Bruke flere variabler/items (som regel spørsmål eller påstander) med høy samvariasjon som analyseres som en felles variabel, slik at effekten av ulik forståelse av hvert enkelt begrep reduseres. En slik gruppe med variabler kalles et konstrukt.

Disse verktøyene har blitt brukt i utformingen av spørreskjemaet.

4.1.1 Digitalt spørreskjema som datainnsamlingsmetode

Robson og McCartan (2017) sin bok er benyttet gjennomgående i dette kapitlet for å vurdere ulike datainnsamlingsmetoder og metodiske problemstillinger. Digitale spørreskjemaer, som er valgt i dette prosjektet, har både fordeler og ulemper sammenlignet med andre datainnsamlingsmetoder. En vesentlig egenskap digitale spørreskjemaer har, er at de effektivt kan nå ut til store mengder respondenter (Robson og McCartan, 2017, s. 251).

Tiden det tar å sende ut og hente inn er lite påvirket av antallet respondenter. De er derfor svært egnet for å gjennomføre kvantitative studier med store grupper. I tillegg bidrar de til at det er enkelt å gjøre datainnsamlingen anonym. Dette er hovedegenskapene som gjorde at det i dette prosjektet ble valgt å lage et digitalt spørreskjema. En typisk utfordring knyttet til digitale spørreskjemaer er blant annet knyttet til at det er en reell risiko for at en lav andel

av de inviterte svarer på spørreskjemaet. Faktorer som bidrar til å øke svarandelen er å lage et spørreskjema som tar forholdsvis kort tid å besvare og å komme med påminnelser og oppfordringer i etterkant av utsendelsen. En annen utfordring er at siden det ikke er anledning for respondentene til å stille oppklarende spørsmål underveis, er det mulig at respondentene tolker spørsmålene på ulike måter og dermed i praksis ikke svarer på det samme, som går ut over resultatenes validitet. Det er heller ikke mulig å kontrollere respondentenes omgivelser; mens noen svarer alene, kan andre gå sammen i grupper og svare, og dermed påvirke hverandres svar. I tillegg krever det kompetanse med relevante IT-verktøy for å utforme og sende ut et spørreskjema, som kan være en utfordring (Robson og McCartan, 2017, s. 249-256).

4.1.2 Utforming av spørreskjema

Jeg valgte å bruke programmet Nettskjema fra Universitetet i Oslo for å lage den tekniske versjonen av spørreskjemaet (se vedlegg A). Det ble valgt at spørreskjemaet ikke innhenter personidentifiserende opplysninger og unngår sensitive eller personlige temaer.

Spørreskjemaet er dermed anonymt og i tråd med retningslinjer for personvern så lenge det ikke er mulig å identifisere deltakere ved å kombinere bakgrunnsinformasjon (NTNU). Jeg lette etter ferdig utviklede og validerte spørreskjemaer innen beregningsorientert fysikk, men det fant jeg ikke. Derfor valgte jeg å utforme spørreskjemaet selv, og jeg tok utgangspunkt i EV-modellen og resultatene fra prosjektoppgaven.

Jeg valgte å utforme spørsmålene som påstander som respondentene forholder seg til ved å uttrykke grad av enighet, siden fenomenene som ble målt dreier seg om holdninger og følelser (Postholm og Jacobsen, 2018, s. 179). Jeg valgte å strukturere svaralternativene etter samme modell som i IMPEL-undersøkelsen: en Likertskala bestående av fem svaralternativer: Helt uenig - 1, litt uenig - 2, verken enig eller uenig - 3, litt enig - 4 og helt enig - 5. Dette er hensiktsmessig siden et sted mellom fem og ni svaralternativer gir mest stabile svar og er enklest å forstå for respondenter (Postholm og Jacobsen, 2018, s. 182). Svaralternativene følger kravene om å være gjensidig utelukkende og å være balanserte (Postholm og Jacobsen, 2018, s. 176). Likertskalaer er mye brukt siden de er enkle å utvikle (Robson og McCartan, 2017, s. 308).

Kriterier

Jeg satte, på bakgrunn av resonnementene ovenfor, følgende kriterier til spørreskjemaet:

- Det skulle ikke ta lengre tid enn 10 minutter å svare.
- Spørsmålene skulle kun bruke forståelige og dagligdagse ord og uttrykk for respondentene (Postholm og Jacobsen, 2018, s. 179).
- For å opprettholde kravet om anonymitet, skulle studentene ikke måtte oppgi bakgrunnsinformasjon som kunne være personidentifiserende.
- I deler av spørreskjemaet skulle påstander inngå i konstrukter, altså grupper med påstander som måler den samme dimensjonen. Det ville bidra til å øke spørreskjemaets pålitelighet (Gliem og Gliem, 2003).
- Deler av spørreskjemaet skulle rette seg mot hvert enkelt fysikkemne studenten hadde hatt for å gi et bilde av utviklingen i løpet av studiet.
- Skjemaet skulle inneholde enkelte frisvarspørsmål (Robson og McCartan, 2017, s. 267).

Spørreskjemaets struktur

Jeg valgte å dele spørreskjemaet inn i fire deler:

Del 1:

(Alle, obligatorisk) Del 1 inneholdt spørsmål om bakgrunnsinformasjon, det vil si linje, kjønn og trinn, etterfulgt av et åpent frisvar-spørsmål om studentens generelle syn på beregningsorientert fysikk.

Trinn, linje og kjønn ble inkludert som obligatoriske spørsmål siden det var viktig informasjon for de påfølgende analysene og for å vite hvilke emner studentene kunne evaluere i senere deler av spørreskjemaet. Her var det viktig å unngå at bakgrunnsinformasjonen kunne være personidentifiserende. Ettersom det er et lavt antall kvinnelige studenter per trinn på BFY, kunne det være en fare for at studenter som krysset av for BFY og var kvinne kunne kjennes igjen. Jeg vurderte det dithen at det ikke var personidentifiserende med kun disse tre opplysningene, men unngikk andre spørsmål om bakgrunnsinformasjon som for eksempel alder og geografisk opprinnelse. Hensikten med et frisvar-spørsmål tidlig i skjemaet var å se hvordan respondenten selv valgte å uttale seg om temaet før han eller hun forholdte seg til konkrete påstander som kunne ha en positiv eller negativ ordlyd.

Del 2:

(Alle, ikke obligatorisk) Del 2 inneholdt påstander som utformer konstrukter knyttet til mestringsforventning, interesseverdi, nytteverdi, kostnad og forhåndskunnskaper, i tillegg til påstanden fra IMPEL-undersøkelsen (tabell 1).

Tabell 2: Påstander på del 2 med tilhørende konstrukter. Nummeret foran påstanden peker til hvilken del det tilhører og hvilken rekkefølge påstandene hadde i spørreskjemaet.

Konstrukt	Påstand
Mestringsforventning	2.3 Jeg forventer at jeg klarer å løse programmeringsoppgaver i fysikk. 2.4 Hvorvidt jeg liker en programmeringsoppgave avhenger av i hvilken grad jeg klarer å løse den. (Reversert)
Interesseverdi	2.1 Jeg er interessert i programmering generelt. 2.2 Jeg er interessert i programmering i fysikkemner.
Nytteverdi	2.6 Programmering er viktig hvis jeg skal bli forsker. 2.7 Programmering er viktig i arbeidslivet.
Kostnad	2.11 Programmeringsoppgaver i fysikk er slitsomme/krevende sammenlignet med andre fysikkoppgaver. 2.12 Det er verdt innsatsen å arbeide med programmeringsoppgaver i fysikk. (Reversert)
Forhåndskunnskaper	2.8 Før jeg begynte på dette studiet, kunne jeg bruke if/else-setninger. 2.9 Før jeg begynte på dette studiet, kunne jeg bruke for/while-løkker. 2.10 Før jeg begynte på dette studiet, kunne jeg bruke numpy-biblioteket i Python.
IMPEL	2.5 Jeg lærer fysikk godt av å programmere.

Alle konstruktene unntatt forhåndskunnskaper baserte seg på EV-modellens fire inkluderte dimensjoner. Jeg utviklet selv påstander innenfor dimensjonene i EV-modellen. Ved å bruke EV-modellen, kunne disse påstandene totalt sett gi et bilde av hva som gjorde studenter motivert for å drive med programmering i fysikk. Påstandene jeg utviklet innen mestringsforventning dreide seg om hvorvidt studentene opplevde at de mestret beregningsorientert fysikk og om dette påvirket på deres interesse for temaet. Interesseverdi dreide seg om hvor interesserte studentene var i programmering både generelt og innen fysikk. Nytteverdi rettet seg mot om studentene tenkte at

beregningsorientert fysikk var noe de har bruk for etter endte studier. Kostnad handlet om hvor slitsomt det var å jobbe med beregningsorientert fysikk og hvorvidt studentene syntes det var verdt innsatsen å ha programmering i fysikk. Forhåndskunnskaper ble også inkludert som konstrukt siden det var varierende hvor mye programmeringsferdigheter studentene hadde med seg fra før studiene, og dette kunne spille inn på deres syn på blant annet programmering. Jeg valgte å ha to påstander som dreide seg om relativt grunnleggende og generelle programmeringsferdigheter (if/else-setninger og for/else-løkker) og en påstand med mer avanserte ferdigheter som var rettet spesifikt mot programmering i fysikk. Påstand 2.4 og 2.11 regnes som såkalte negativt ladede påstander, som innebar at et positivt ladet svar ville tilsvare en lav verdi på Likert-skalaen for disse påstandene. Bruk av både positive og negative items er et typisk virkemiddel i spørreskjemaer med bruk av Likert-skala (Robson og McCartan, 2017, s. 309).

Del 2 inneholdt også en påstand om programmering og læring som var hentet fra en spørreundersøkelse som inngikk i IMPEL-prosjektet høst 2019 ved Universitet i Oslo (påstand 2.5 i tabell 2). Denne påstanden ble stilt til de daværende førsteårsstudentene på MTFYMA, BFY og et femårig lektorprogram i realfag, hvilket tilsvarende de nåværende tredjeårsstudentene. I IMPEL-spørreundersøkelsen var det 158 deltakere, og påstanden fikk en gjennomsnittsscore på 2,6 og et standardavvik på 1,0. Hensikten var å kunne sammenligne disse resultatene med resultatene i denne studien for å se om studentene hadde endret forståelse av hvordan de lærer fysikk av å programmere.

Del 3:

(Andre- og tredjeårsstudenter, ikke obligatorisk) Del tre inneholdt fire påstander som knyttet til hvert enkelt fysikkemne studentene hadde hatt. Påstandene rettet seg mot hvorvidt studentene hadde nok forutsetninger for å løse programmeringsoppgavene i emnet og hva de syntes om programmeringsoppgaver sammenlignet med vanlige regneoppgaver. Førsteårsstudentene hadde ennå ikke fullført noen emner på NTNU svarte derfor ikke på denne delen av spørreskjemaet.

Tabell 3: De fire påstandene som studentene skulle forholde seg til for hvert enkelt fysikkemne de hadde hatt. Nummeret foran påstanden peker til hvilken del det tilhører og hvilken rekkefølge påstandene hadde i spørreskjemaet.

Påstand
3.1 Jeg hadde fått god nok opplæring i programmering til å løse programmeringsoppgavene.
3.2 Programmeringsoppgavene krevde numerikkferdigheter som jeg ikke hadde lært.
3.3 Jeg mestret programmeringsøvingene bedre enn regneøvingene.
3.4 Jeg foretrakk regneøvinger fremfor programmeringsøvinger.

Som vist i intervjuene fra forprosjektet, var det variasjon fra emne til emne i hva studentene syntes om de beregningsorienterte oppgavene. Ved å se på hvert emne separat kunne man få bedre innblikk i hvordan studenter forholdt seg til programmering i fysikk i ulike stadier av studiet. Påstandene sammenlignet programmeringsoppgavene med de vanlige fysikkoppgavene for å gi grunnlag for å sammenligne. De var konstruert for å identifisere studentenes syn på opplæringen de hadde fått, deres opplevde mestring og deres preferanser.

Del 4:

(Alle, ikke obligatorisk) Del 4 inneholdt et frisvar-spørsmål om respondentene hadde noe de ønsket å legge til angående programmering i fysikk eller spørreskjemaet. Deretter ble studentene invitert til å delta på intervju senere på semesteret. Studentenes anonymitet ble ivaretatt da denne påmeldingen foregikk uavhengig av spørreskjemaundersøkelsen.

4.1.3 Utsending av spørreskjema

Datainnsamlingens forløp gikk i utgangspunktet ut på å sende ut spørreskjemaene på mail. I Nettskjema legger man inn en eller flere mailadresser i programmet, og mottakeren mottar en automatisk generert invitasjon via mail. For å få tilgang til mailadressene kontaktet jeg studieadministrasjonen som ga meg tilgang til maillistene for de aktuelle studentene.

Majoriteten av studentene mottok spørreundersøkelsen samtidig, det eneste unntaket er de studentene i tredje årstrinn på MTFYMA som går matematikkretningen (de utgjør vanligvis omtrent 40 % av tredjeårsstudentene), de mottok invitasjonen to uker senere enn resten av studentene. Jeg valgte å holde undersøkelsen åpen i tre uker totalt tidlig på semesteret (i

september). I tillegg valgte jeg og mine veiledere å oppsøke studenter og dele ut informative papirer i deres læringsomgivelser for å gjøre flere oppmerksomme på undersøkelsen. Studentene mottok også en påminnelse på mail. De fleste respondentene svarte i løpet av en uke etter utsending av spørreskjema og en uke etter påminnelse.

4.1.4 Populasjon og utvalg

Siden problemstillingen i dette prosjektet retter seg mot studenters forhold til programmering i fysikk i ulike ledd av ferdighetsstrengen, har jeg valgt å inkludere alle studenter som følger ferdighetsstrengen dette semesteret. Populasjonen er derfor alle studenter i første, andre og tredje årstrinn på de to studieprogrammene MTFYMA (Fysikk og matematikk – sivilingeniør) og BFY (Bachelor i fysikk). Siden ferdighetsstrengen ble introdusert høsten 2019, er det kun disse studentene som har fulgt den helt fra studiestart.

Tabell 4: Antall studenter fordelt på linje og trinn som har henholdsvis mottatt spørreundersøkelsen, svart på spørreundersøkelsen og til slutt andelen studenter som har svart.

	Trinn	MTFYMA	BFY	TOTAL
Totalt antall studenter	1. årstrinn	120	48	168
	2. årstrinn	120	36	156
	3. årstrinn	109	42	151
	TOTAL	349	126	475
Antall studenter som har svart	1. årstrinn	38	19	57
	2. årstrinn	24	10	34
	3. årstrinn	23	14	37
	TOTAL	85	43	128
Andel studenter som har svart (%)	1. årstrinn	31,7	39,6	33,9
	2. årstrinn	20,0	27,8	21,8
	3. årstrinn	21,1	33,3	24,5
	TOTAL	24,4	34,1	26,9

Totalt utgjør populasjonen 475 studenter som mottok spørreskjemaet (tall fra studieadministrasjonen). 128 studenter besvarte ved å fylle ut spørreskjemaet, og disse utgjør utvalget, som tilsvarer 26,9 prosent (tabell 1) av de spurte. Ved å sammenligne årstrinnene ser man at det var ganske jevn svarandel blant studentene i andre og tredje

årstrinn, mens i første årstrinn var svarandelen omtrent 10 prosentpoeng høyere, til tross for at det var omtrent like mange studenter på hvert trinn. Det var dessuten en betydelig høyere svarandel fra BFY enn fra MTFYMA, med en forskjell på omtrent 10 prosentpoeng på hvert trinn.

4.1.5 Analyse av spørreskjema

Kvantitativ dataanalyse

Dataene ble kodet inn og analysert med SPSS 27. Deretter ble variablene klassifisert som enten nominale eller ordinale, og de ordinale variablene ble kodet til tall med følgende oversettelse: Helt uenig – 1, litt uenig – 2, verken enig eller uenig – 3, litt enig – 4, helt enig – 5. Ved hjelp av bivariat analyse ble spørsmålene presentert i stolpediagrammer som avhengig variabel av årstrinn, der diagrammene viste gjennomsnittsverdier og standardavvik. Videre ble de forhåndsdefinerte konstruktene fra EV-modellen testet ved å måle deres interne konsistenspålitelighet ved variabelen Chronbachs alfa (videre referert til som alfa, se tabell 5) (Gliem og Gliem, 2003). I den forbindelse ble svarskalaen til påstand 2.4 og 2.12 reversert for at påstandene skulle ha samme ladning som sine overordnede konstrukter. Videre brukte jeg SPSS til å regne ut en tabell med Pearson-korrelasjonen mellom spørsmål og konstrukter. Pearson-korrelasjon går fra -1 til 1, der -1 betyr perfekt negativ korrelasjon, 0 betyr ingen korrelasjon og 1 betyr perfekt positiv korrelasjon. Signifikansnivået var 0,05 med to-halet t-test, altså fulgte den Fisher-kriteriet (Field, 2009, s. 49-51). For å finne tilsvarende tabell for hvert enkelt årstrinn, ble dataene lagt inn i SPSS med hvert årstrinn hver for seg.

Kvalitativ dataanalyse av frisvarspørsmål

Spørreskjemaet inneholdt to frisvarspørsmål. Det første frisvarspørsmålet var et obligatorisk spørsmål i begynnelsen av spørreskjemaet: «Hvordan fungerer programmering i fysikkemner for deg?» Det andre frisvarspørsmålet kom på slutten av spørreskjemaet og var frivillig: «Har du noen kommentarer til denne spørreundersøkelsen eller andre kommentarer angående programmering i fysikk?» Det ble gjort en tematisk analyse (Braun og Clarke, 2006) ved hjelp av analyseprogrammet NVivo, tematisk analyse blir beskrevet mer detaljert i delkapittel 4.2.6. De to spørsmålene ble analysert hver for seg, og svarene ble fordelt etter årstrinn. Analysen foregikk ved å lese gjennom frisvarene flere ganger og bli kjent med datamaterialet, samle og gruppere svarene i koder og til slutt samle kodene i større

kategorier, som kalles temaer. Siden de to spørsmålene som ble stilt var svært åpne, var det stor variasjon i svarene og det ble i hovedsak brukt en induktiv tilnærming i utformingen av koder. I frisvars spørsmål 2 var svarene såpass få og kodene såpass forskjellige at jeg vurderte det som lite hensiktsmessig å dele dem inn i temaer.

4.1.6 Validitet spørreskjemaundersøkelse

Med validitet menes i hvilken grad en undersøkelse faktisk måler det den er ment å måle. Et underliggende kriterium for validitet er om undersøkelsen er pålitelig. Pålitelighet kan beskrives som i hvilken grad man får de samme resultatene om man gjennomfører den samme undersøkelsen flere ganger med ulike respondenter (Field, 2009, s. 11-12; Robson og McCartan, 2017, s. 19). Siden denne spørreundersøkelsen kun er gjennomført en gang, er dens grad av pålitelighet i utgangspunktet lav, og dens validitet må vurderes med dette forbeholdet. Men, siden studenter i både andre og tredje årstrinn har svart på de samme spørsmålene med tanke på førsteårsemenene, vil en sammenligning av disse gi en pekepinn på hva studenter i ulike trinn tenker om de samme emnene. En annen kilde til pålitelighet er at flere av påstandene er delt inn i konstrukter, siden eventuelle ulike tolkninger av enkeltspørsmål jevnes ut.

Validitet kan deles i to kategorier: intern og ekstern. Intern validitet handler om sammenhengen mellom behandling og utfall, mens ekstern validitet handler om undersøkelsens generaliserbarhet (Robson og McCartan, 2017, s. 107-109).

Intern validitet

Hovedkomponenten i spørreundersøkelsens interne validitet handler om hvorvidt spørsmålene og påstandene er entydige. For å bidra til dette ble flere av påstandene delt inn i konstrukter. Et konstrukt av består av variabler som skal måle den samme dimensjonen, forskjellen mellom påstandene er ulik formulering. For å vurdere om påstandene i et konstrukt faktisk har målt det samme, brukes Cronbachs alfa. Cronbachs alfa er et mål på intern konsistenspålitelighet (internal consistency reliability) som varierer mellom 0 og 1. Jo nærmere den er 1, jo høyere konsistens er det mellom de målte variablene.

Tommelfingerregelen sier at laveste grense for en akseptabel verdi for Chronbachs alfa ligger mellom 0,6 og 0,7 (tabell 5) (Gliem og Gliem, 2003).

Tabell 5: Konstrukter med tilhørende påstander og Cronbachs alfa, som ble utregnet ved hjelp av analyseprogrammet SPSS.

Konstrukt	Påstand	Chronbachs alfa
Mestringsforventning	2.3 Jeg forventer at jeg klarer å løse programmeringsoppgaver i fysikk. 2.4 Hvorvidt jeg liker en programmeringsoppgave avhenger av i hvilken grad jeg klarer å løse den. Reversert	0.453
Interesseverdi	2.1 Jeg er interessert i programmering generelt. 2.2 Jeg er interessert i programmering i fysikkemner.	0.777
Nytteverdi	2.6 Programmering er viktig hvis jeg skal bli forsker. 2.7 Programmering er viktig i arbeidslivet.	0.687
Kostnad	2.11 Programmeringsoppgaver i fysikk er slitsomme/krevende sammenlignet med andre fysikkoppgaver. 2.12 Det er verdt innsatsen å arbeide med programmeringsoppgaver i fysikk. Reversert	0.487
Forhåndskunnskaper	2.8 Før jeg begynte på dette studiet, kunne jeg bruke if/else-setninger. 2.9 Før jeg begynte på dette studiet, kunne jeg bruke for/while-løkker. 2.10 Før jeg begynte på dette studiet, kunne jeg bruke numpy-biblioteket i Python.	0.875
IMPEL	2.5 Jeg lærer fysikk godt av å programmere.	

Ved å plassere alle dataene i det statistiske analyseprogrammet SPSS, ble Cronbachs alfa eller bare alfa regnet ut for hvert enkelt konstrukt (tabell 5). Påstand 2.4 regnes som en negativ påstand og har derfor reversert svarskala under konstruktanalysen. Påstand 2.12 har også reversert svarskala siden kostnad er et negativt konstrukt og 2.12 er en positivt ladet påstand. Blant de fem forhåndsdefinerte konstruktene, er det kun tre som har en alfa som er høy nok til at man kan si at påstandene som inngår i konstruktet faktisk måler den samme dimensjonen (markert med grønt). De tre validerte konstruktene er interesseverdi, nytteverdi og forhåndskunnskaper, mens de to konstruktene med for lav alfaverdi er mestringsforventning og kostnad (markert med rødt). En mulig årsak til dette er at en kombinasjon av positivt og negativt ladde påstander fører til mindre samvariasjon mellom påstander, siden begge konstruktene med lav alfa består av påstander med ulike ladninger. Ideelt sett burde spørreskjemaundersøkelsen vært testet i forkant, og formuleringer rettet hvis ikke de utgjorde et konstrukt. Videre vil påstand 2.3 og 2.4 samt 2.11 og 2.12 bli vurdert som separate variabler i analyseseksjonen. Dette reduserer påstandenes pålitelighet noe.

Påstand 2.3 antas å være den påstanden som utgjør det mest direkte målet for mestringsforventning, mens påstand 2.4 heller sees på som en kobling mellom mestring og interesse. Tilsvarende for kostnadskonstruktet er påstand 2.11 den påstanden som antas å måle kostnaden best, her sammenlignet med vanlige fysikkoppgaver. Påstand 2.12 måler heller koblingen mellom kostnad og for eksempel nytteverdi.

Påstandene fra del 2 fikk stort sett en score som lå over midten, uavhengig av om spørsmålene var positivt eller negativt ladet. Del 3, på en annen side, hadde like mange høye svar som lave svar. I del 2 svarte også førsteårsstudentene, men ikke i del 3. Det var langt flere førsteårsstudenter i utvalget enn andre- og tredjeårsstudenter. Sammenlagt tyder dette på at spesielt førsteårsstudentene hadde en høy tendens til å si seg enig med påstandene. En mulig årsak til dette er at førsteårsstudentene hadde svakere formeninger om hva de skulle svare på påstandene siden de så vidt hadde begynt å studere da de svarte på spørreskjemaet.

Spørreskjemaet hadde også en del hvor det ble spurt om studentenes karakterer, men det kom fram i frisvarspørsmålet på slutten av spørreundersøkelsen at flere av eksamenene hadde godkjent/ikke godkjent vurderingsform, hvilket spørreskjemaet ikke tok høyde for. Siden det kommer fram av frisvaret at dette medførte at studentene besvarte karakterspørsmålet på ulike måter (noen tenkte at godkjent var det samme som en C, andre lot være å svare) regnes påliteligheten til disse påstandene for å være for lav til å bli inkludert i analysene.

Ekstern validitet/generaliserbarhet

Et sentralt mål på generaliserbarhet er i hvilken grad utvalget er representativt for populasjonen det representerer (Robson og McCartan, 2017, s. 19). Med et utvalg på 128 studenter (tabell 4), kan det være mye informasjon å hente, men siden det kun utgjør 26,9 prosent av den totale populasjonen og utvalget er selvselektert (ikke tilfeldig valgt), kan en ikke si at utvalget er representativt for populasjonen. Mangel på representativitet er nødvendigvis alltid en problemstilling i samfunnsundersøkelser der kun deler av populasjonen deltar, siden all deltakelse er frivillig. Det kan i utgangspunktet virke som om utvalget er helt tilfeldig, siden det ikke er noen tiltenkt systematikk i hvordan respondentene er valgt. På en annen side er det ingen garanti for at det ikke foreligger fellesnevner blant de som valgte å svare som ikke er representative for hele populasjonen. Det kan for

eksempel tenkes at det kun er de studentene som mestrer programmering i fysikk godt som svarer, eller motsatt, at det er de som sliter som har et behov for å gi uttrykk for misnøye eller strev. Det er også mulig at det ikke er nivået som er avgjørende for om en student velger å svare eller ikke, og at det er andre faktorer i stedet som er avgjørende. Det er vanskelig å vite om man har eliminert en slik skjevhet i utvalget uten å ha en svært høy svarprosent, men høy svarprosent er imidlertid vanskelig å oppnå i en digital spørreundersøkelse (Robson og McCartan, 2017, s. 282-283).

En måte å se etter skjevheter i utvalget er å se på antallet besvarelser i de ulike undergruppene. Tallene fra de enkelte undergruppene viser at det er langt flere førsteårsstudenter som svarte enn fra øvrige trinn. En mulig årsak til dette er at den fysiske rekrutteringskampanjen viste seg å i hovedsak bidra til å rekruttere førsteårsstudenter. Denne rekrutteringen kan også ha skapt en skjevhet i og med at den kun nådde ut til studenter som oppholdt seg på campus. Utslagene dette får på spørreundersøkelsen vil kunne oppdages ved å analysere resultatene for hver undergruppe separat. Den ujevne kjønnsfordelingen tyder også i retning av en skjevhet i utvalget, men årsaken til dette er uvisst. Konsekvensene av det vil det derimot også være mulig å analysere ved å undersøke svarfordelingene for hvert kjønn, dette vil gi et bilde av hvorvidt kjønnsbalansen har påvirket resultatene.

Sikre faktorer som påvirket utvalget, var om studentene oppdaget invitasjonen eller påminnelsen i mailboksen sin innen svarfristen og om de støtte på meg eller mine veiledere når vi gikk rundt på campus for å oppfordre studentene til å svare. Usikre faktorer som kan ha påvirket utvalget er om studentene syntes temaet for spørreskjemaet var interessant nok til å svare, om studentene mottok for mange spørreskjemaer på mail til at de ville velge å svare på alle, om spørreskjemaet var for langt, irrelevant eller eventuelt andre ting som kan ha ført til at de lot å være å fullføre besvarelsen.

Tabell 6: Utvalg som inkluderer dimensjonen kjønn.

		Kvinne	Mann	Annet	TOTAL
BFY	1. årstrinn	1	18	0	19
	2. årstrinn	1	9	0	10
	3. årstrinn	4	10	0	14
MTFYMA	1. årstrinn	15	23	0	38
	2. årstrinn	10	13	1	24
	3. årstrinn	13	10	0	23
TOTAL		44	83	1	128

Respondentene måtte også oppgi kjønn. Tabell 6 viser en betydelig større kvinneandel hos MTFYMA enn hos BFY, som er å forvente siden det generelt er flere kvinner på MTFYMA enn på BFY. Det tabellen også viser er at kvinneandelen blant respondentene øker for hvert årstrinn på begge studieprogrammer, selv om det ikke er noe som tyder på at kjønnsfordeling i populasjonen endrer seg i løpet av de tre første studieårene. Det er ikke mulig å få opplyst den faktiske kjønnsfordelingen i populasjonen. Tredje årstrinn MTFYMA er den eneste undergruppen med flere kvinner enn menn blant respondentene. Totalt var det om lag dobbelt så mange mannlige studenter som kvinnelige studenter som svarte på spørreundersøkelsen.

4.2 Kvalitativ metode

Dette kapitlet har fokus på intervjuene som jeg gjennomførte i etterkant av det digitale spørreskjemaet i den hensikt å få en dypere kvalitativ forståelse for resultatene fra den kvantitative analysen.

4.2.1 Kvalitative intervjuer

Intervjuene var semistrukturerte, som innebar at jeg fulgte en intervjuguide som fungerte som sjekklister (Robson og McCartan, 2017, s. 285-286). Selv om jeg fulgte intervjuguiden, kunne jeg som intervjuer være fleksibel med tanke på den presise formuleringen av hvert spørsmål og rekkefølgen spørsmålene kom i. Semistrukturerte intervjuer brukes for å forstå deltakerens perspektiv (Postholm og Jacobsen, 2018, s. 121).

Intervjuene var individuelle, som gjorde det mulig å ha en relativt lang intervjuguide og samtidig la hver enkelt intervjudeltaker gå i dybden på hvert enkelt spørsmål. Til gjengjeld reduserte dette antallet studenter det ble mulig å intervjuer uten å få svært store datamengder, men det ble ansett å være passende for den kvalitative undersøkelsen siden hensikten var å gå i dybden på de kvantitative dataene fra spørreundersøkelsen.

I forkant av intervjuene ble prosjektet meldt til og godkjent av Norsk senter for forskningsdata (vedlegg D).

4.2.2 Utforming av intervjuguide

Ved valg av spørsmål å inkludere i intervjuguiden (vedlegg B) var det viktig å rette seg mot problemstillingen og resultatene fra spørreundersøkelsen. Det innebar å prøve å rette samtalen inn mot dimensjonene i EV-modellen og å danne et bilde av hvordan deltakerens inntrykk av programmering i fysikk har utviklet seg i løpet av studiet.

Det første spørsmålet er et åpent spørsmål om studentens bakgrunn og erfaring med programmering før han eller hun begynte på studieprogrammet, for å skaffe relevant bakgrunnsinformasjon for andre temaer. Deretter går guiden inn på studentens generelle inntrykk av programmering og ber om kjennetegn på en god programmeringsoppgave i fysikk. Disse spørsmålene er relativt åpne og tillater deltakeren å trekke inn selvvalgte elementer og temaer. Etterpå retter spørsmålene seg mer konkret mot EV-modellen ved å spørre om motivasjon og nytte.

Siden fysikkemnene i hovedsak består av regneoppgaver som løses på papir, er det de oppgavene som kan sies å sette standarden for fysikkemnene med tanke på for eksempel vanskelighetsgrad og arbeidsmengde. Derfor inneholder intervjuguiden en rekke spørsmål som søker å sammenligne programmeringsoppgavene i fysikk med de vanlige regneoppgavene.

Siste del av intervjuguiden inneholder to spørsmål som søker å sette bestemte fysikkemnene i fokus, som bidrar til å gi et detaljert bilde av studentenes programmeringserfaringer i ulike ledd av ferdighetsstrengen. Aller sist følger et helt åpent spørsmål, for å gi deltakeren mulighet til å legge til relevante ting som det ikke var anledning for tidligere i intervjuet.

4.2.3 Utvalg av intervjudeltakere

Utvalget er selvselektert fra studentgruppen som deltok i spørreundersøkelsen, det vil si alle studenter i første, andre og tredje trinn ved studieprogrammene MTFYMA og BFY.

Rekrutteringsprosessen til intervjuene foregikk ved at det på slutten av spørreskjemaundersøkelsen forelå en invitasjon til å stille til intervju senere på semesteret. Dette foregikk uavhengig av spørreskjemaet.

Totalt var det tre studenter som ønsket å stille til intervju, to i andre årstrinn og en i tredje årstrinn, alle på studieprogrammet MTFYMA. Disse tre utgjør utvalget i den kvalitative intervjuundersøkelsen.

4.2.4 Intervjuenes forløp

Jeg gjennomførte intervjuene i november, altså omtrent to måneder etter at studentene fylte ut spørreskjemaet. I forkant av og ved oppmøte til intervjuet ble studentene tilsendt samtykkeerklæring (vedlegg C), som de ble bedt om å lese, avklare spørsmål og underskrive. De gav også muntlig samtykke til å ta opp intervjuet med en lydopptaker. Det var planlagt at intervjuene skulle vare i omtrent 30 minutter, i praksis varierte det mellom 27 og 55 minutter.

Videre gikk vi gradvis gjennom intervjuguiden (vedlegg B) i den rekkefølgen det var satt opp, i alle tre intervjuer. I mange tilfeller stilte jeg oppfølgingsspørsmål og inngående spørsmål innimellom for å få deltakeren til å utdype eller forklare nærmere.

4.2.5 Validitet intervjuundersøkelse

Hovedutfordringen i den kvalitative undersøkelsen med tanke på validitet er knyttet til den eksterne validiteten, altså det at utvalget var lite og selvselektert. Det var kun studentene som hadde fullført spørreundersøkelsen som hadde mulighet til å melde seg til intervju. Dermed tilhører utvalget i intervjuene også utvalget i spørreundersøkelsen og kan antas å ha den samme skjevheten sammenlignet med den totale populasjonen, som kan være forårsaket av for eksempel stor interesse eller sterke meninger knyttet til programmering i fysikk. Et utvalg på kun tre studenter er lite og representerer kun de tre, men den kvalitative innsikten deres erfaringer gir, har potensial å gi verdifull innsikt i problemstillingen.

Den interne validiteten er i hovedsak utfordret av at jeg som intervjuer kan ha påvirket svarene til deltakerne gjennom mitt kroppsspråk og valg av formuleringer, spørsmål og

kommentarer (Robson og McCartan, 2017, s. 287). I et enkeltintervju er ikke deltakeren påvirket av andre deltakere, men det øker sannsynligheten for at jeg som intervjuer har påvirket svarene. Som intervjuer er det viktig å være lyttende og å unngå å prate for mye, men samtidig er en viss deltakelse av verdi for å forstå deltakerne gjennom å ha samme bakgrunn og å hjelpe dem i dialogen. Siden både jeg og intervjudeltakerne går på studieprogrammet MTFYMA, kunne jeg ofte kjenne meg igjen i deltakernes beskrivelser og stille detaljerte oppfølgingsspørsmål.

4.2.6 Analyse av intervjuer

Opptakene fra intervjuene ble overført fra opptakerne til en passordbeskyttet Onedrive-fil og deretter transkribert. For å opprettholde deltakernes anonymitet, brukes følgende anonyme kodenavn i rapporten: M2, K2 og K3. M står for mann, K står for kvinne og tallet antyder studentens årstrinn. All informasjon om deltakerne er også lagret i Onedrive og vil bli slett ved slutten av prosjektet. Under transkripsjonen ble lyder som «Eh» og «Mm» inkludert der jeg mente de var av betydning for å forstå ordlyd eller uttrykke en mening. Alt er skrevet i bokmål, også der hvor eventuelle dialektord er blitt brukt.

Jeg foretok en tematisk analyse av intervjuene med analyseprogrammet NVivo. Analysen fulgte følgende sekstrinnsmodell (Braun og Clarke, 2006):

1. Bli kjent med dataene.

Denne delen ble i hovedsak gjennomført i gjennomføringen og transkriberingen av intervjuene, siden jeg gjorde begge deler selv. I etterkant av transkripsjonen hadde jeg fått et godt overblikk over innholdet i de tre intervjuene.

2. Lage startkoder.

Denne prosessen gikk ut på å lese gjennom intervjuene og samle utdrag som tar opp de samme områdene i koder, altså grupper med korte, beskrivende titler. I denne prosessen vekslet jeg mellom en induktiv og en deduktiv tilnærming, avhengig av om tekstutdragene beskrev elementer som kan knyttes direkte til problemstillingen og forskningsspørsmålene, det vil si de fire dimensjonene i EV-modellen (mestringsforventning, interesseverdi, nytteverdi og kostnad) og utvikling, eller om de beskrev andre elementer som allikevel er relevant for undersøkelsen.

3. Lete etter temaer.

Når alle kodene var lagd, lette jeg etter overordnede fellesnevnerne som kodene kunne kategoriseres under. Som i punkt 2 ble det vekslet mellom en induktiv og en deduktiv tilnærming etter behov, først ble det lagd temaer som kunne knyttes til tidligere forskning og forskningsspørsmål, deretter ble det lagd temaer som fanget opp kodene som ikke passet inn i kjente kategorier. I noen tilfeller ble det også utviklet undertemaer, spesielt i forbindelse med å dimensjonene i EV-modellen.

4. Gå gjennom og moderere temaene gjentatte ganger.

Det ble i etterkant av den første temainndelingen gjennomført mindre endringer, blant annet ble et undertema fjernet for å forenkle strukturen. De tilhørende kodene ble lagt under det overordnede temaet i stedet.

5. Definere de endelige temaene.

Til slutt ble temainndelingen slik at det er tre temaer (forhåndskunnskaper og -forventninger, motivasjon og progresjon) som er resultat av en deduktiv tilnærming der temaene står i direkte stil med forskningsspørsmål og resultater fra spørreskjemaet. De to andre temaene (systemet rundt studentene og kjennetegn på en god programmeringsoppgave) er basert på en induktiv tilnærming og tar opp andre vinklinger og perspektiver enn det som er gjort tidligere. Det er fem temaer fra den tematiske analysen: forhåndskunnskaper og -forventninger, motivasjon, progresjon, systemet rundt studentene og kjennetegn på en god programmeringsoppgave. Temaet motivasjon er delt inn i fire underkategorier: mestringsforventning, interesseverdi, nytteverdi og kostnad.

Forhåndskunnskaper og -forventninger dreier seg om kunnskapene og forventningene studentene hadde før de begynte å studere angående programmering. Motivasjon retter seg mot dimensjonene i EV-modellen og forsøker å kartlegge studentenes generelle syn på beregningsorientert fysikk i deres nåværende situasjon. Progresjon ser på studentenes beskrivelser av hvordan deres erfaringer med programmering i fysikk har utviklet seg fra begynnelsen av studiet til der de er nå. Dette kan også sees i lys av EV-modellen. Systemet rundt studentene omhandler studentenes beskrivelser av studieprogrammet sitt med tanke på for eksempel emner, forelesninger og øvinger. Det siste temaet er kjennetegn på en god

programmeringsoppgave, hvor studentene går inn på hvilke deler av programmeringsoppgavene i fysikk de synes har vært spesielt gode.

Tabell 7: Temaer og undertemaer fra den tematiske analysen av intervjuene.

Tema	Undertema
Forhåndskunnskaper og -forventninger	
Motivasjon	Mestringsforventning
	Interesseverdi
	Nytteverdi
	Kostnad
Progresjon	
Systemet rundt studentene	
Kjennetegn på en god programmeringsoppgave	

6. Rapportere resultatene.

Resultatene og diskusjonen av den tematiske analysen er i kapittel 5 og 6. I kapittel 6 drøftes analysen av intervjuene sammen med analysen av spørreskjemaundersøkelsen.

4.3 Metodediskusjon

4.3.1 Studiets design

Denne undersøkelsen benytter mixed methods ved å bestå av en kvantitativ tverrsnittsundersøkelse med digitalt spørreskjema i september og en kvalitativ intervjuundersøkelse i november. Hensikten med fremgangsmåten var å først finne erfaringene til en stor gruppe studenter og deretter gå i dybden på disse erfaringene med enkeltstudenter. Designets format var hensiktsmessig for min masteroppgave siden det tillot meg å gjennomføre begge undersøkelsene i løpet av høsten masteren ble skrevet. På en annen side hadde det vært mer ideelt å kunne følge den samme årgangen over en lengre tidsperiode, siden det ville gitt bedre grunnlag for å trekke slutninger om årsak og virkning (Postholm og Jacobsen, 2018, s. 80-81).

Konsekvensene av mixed methods har blant annet vært at datainnsamlingsprosessen totalt sett har vært omfattende, siden den inneholdt to ulike deler. Ved å rekruttere intervjudeltakere gjennom spørreskjemaundersøkelsen, ble undersøkelsene mer effektive, siden en stor og uforutsigbar del av arbeidet er å rekruttere forskningsdeltakere. En ulempe

med å rekruttere intervjudeltakere på denne måten er at det øker skjevheten i utvalget. Noe som kan være enten en fordel eller en ulempe med å ha to datainnsamlingsmetoder, er at datamaterialet kan bli stort. Det kom tydelig fram i analysearbeidet at det heller ville bli et problem å ha for mange resultater å analysere enn for lite. Dette har stilt høyere krav til meg som forsker med å ignorere det som er mindre viktig og å fokusere på de delene av analysene som bidrar best til å besvare problemstillingen og forskningsspørsmålene.

4.3.2 Spørreskjemaundersøkelse

I mangel av tidligere spørreskjemaer som omhandler beregningsorientert fysikk, lagde jeg spørsmålene selv. Spørsmålene skulle inneholde dimensjoner fra EV-modellen gjennom å gruppere spørsmålene i konstrukter. Men som vist i konstruktanalysen, hadde ikke konstruktene mestringsforventning og kostnad høy nok verdi av chronbachs alfa til å kunne anses som konstrukter. Jeg har derfor valgt å la enkeltspørsmål representere disse dimensjonene i analysen og diskusjonen, med forbehold om at validiteten til disse dimensjonene blir mer usikker.

Funksjonene i programmet SPSS har påvirket hvilke analyser som ble gjort og hvordan resultatene ble presentert. Alt jeg kunne om SPSS lærte jeg meg på egenhånd underveis i arbeidet med masteroppgaven. Arbeidet med spørreskjemaundersøkelsen var preget av at jeg ikke hadde noe særlig erfaring med det fra før, som medførte at jeg hadde mye å sette meg inn i underveis og begrensede forventninger til utfallet av undersøkelsen angående blant annet svarandel, tolkning og analyse av kvantitativ data kombinert med kvalitative data.

4.3.3 Intervjuundersøkelse

I forbindelse med fordypningsprosjektet mitt våren 2021, gjennomførte jeg intervjuer med studenter som en innledende studie av studentenes erfaringer knyttet til beregningsorientering i fysikk. Dermed hadde jeg et godt forhåndsinntrykk av hva undersøkelsen innebar, ville bringe og hva det var interessant å gå nærmere inn på. Dette gjorde det blant annet enklere å planlegge hvor mye tid analysene ville ta. Allikevel endte masterprosjektet opp med å følge planlagt tidsskjema. Til forskjell fra forprosjektet ble den tematiske analysen gjort i programmet NVivo. I forprosjektet ble analysen gjort for hånd. Den tematiske analysen gikk raskere i masterprosjektet, som jeg tror skyldes både at jeg brukte NVivo og at jeg var mer kjent med metoden, tematikken og caset. I masterprosjektet

utgjorde den tematiske analysen avslutningen på undersøkelse som ble startet i fordypningsprosjektet. NVivo gjorde at all tekstanalyse ble gjort digitalt, som gjorde det enklere å ha oversikt over datamaterialet og kodeinndelingen. Til gjengjeld kan den raske og effektive analyseprosessen i NVivo ha gitt mindre nærhet til datamaterialet enn hvis den hadde vært gjort for hånd på papir.

5 Resultater

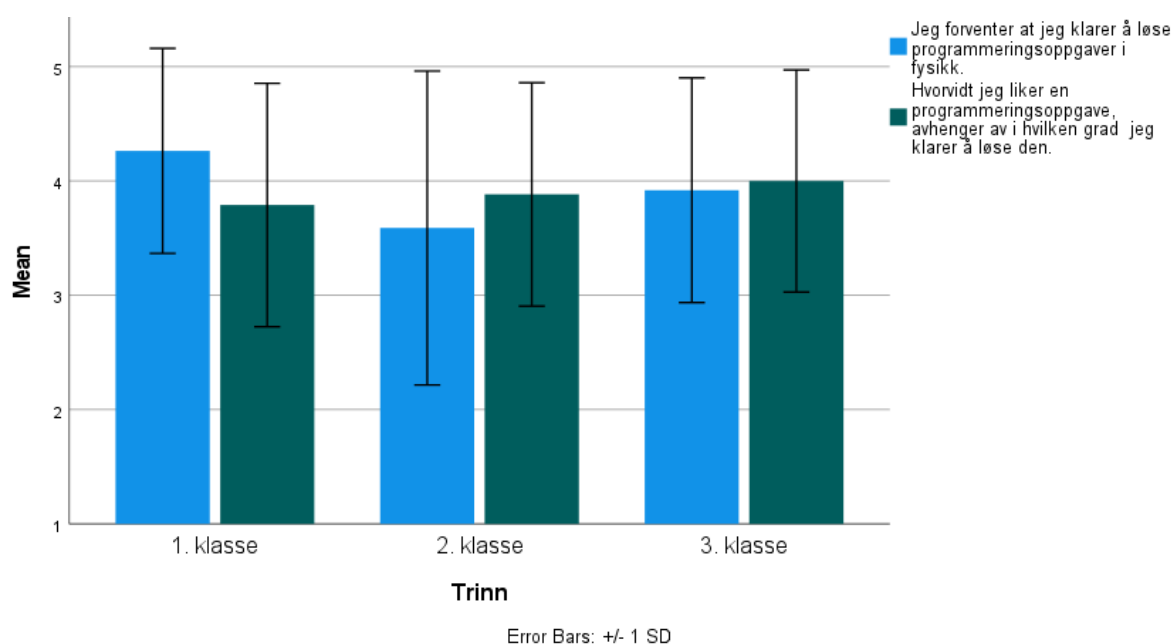
5.1 Spørreundersøkelse

Spørreskjemaet bestod av fire deler der del en og fire brukte et kvalitativt svarformat i form av frisvarsspørsmål, og del to og tre brukte et kvantitativt svarformat ved å benytte en Likertskala som går fra 1 – Helt uenig til 5 – Helt enig.

Denne seksjonen tar for seg diagrammer som er produsert med SPSS (med unntak av de kvalitative svarene på frisvarsspørsmålene), samt forklarende og utdypende tekst. Den forklarende teksten inneholder studentsitater hentet fra frisvarsspørsmålene og egne kommentarer. Studentenes svar er illustrert med søylediagrammer hvor lengden på søylene tilsvarer gjennomsnittlig score (s) og den svarte linjen som delvis dekker søylen er standardavviket (d).

Første del av denne seksjonen inneholder påstandene fra del 2 i spørreskjemaet, hvor påstandene er gruppert etter de forhåndsdefinerte konstruktene mestringsforventning, interesseverdi, nytteverdi, kostnad og forhåndskunnskaper.

Mestringsforventning



Figur 5: Svar på spørsmålene fra konstruktet mestringsforventning.

Figur 5 viser påstandene fra konstruktet mestringsforventning: «Jeg forventer at jeg klarer å løse programmeringsoppgaver i fysikk.» (blå søyle) og «Hvorvidt jeg liker en programmeringsoppgavene, avhenger av i hvilken grad jeg klarer å løse den.» (grønn søyle).

Studentene har høyest forventning om å klare å løse programmeringsoppgaver i fysikk i 1. klasse (uttrykt kun ved blå søyle på grunn av konstruktanalysen, $s=4,2$, $d=0,9$).

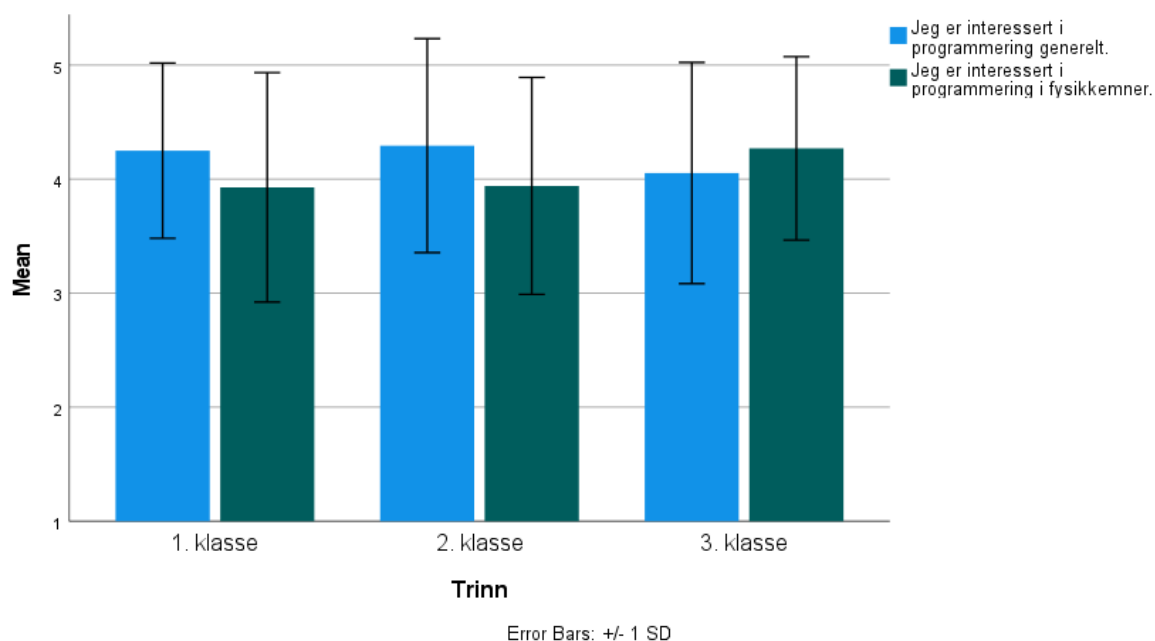
Mestringsforventning er derimot ganske mye lavere for 2. klasse ($s=3,6$), selv om gjennomsnittet er godt over midtverdien. I 2. klasse er også standardavviket størst ($d=1,4$). Deretter øker mestringsforventningen til studentene igjen i 3. klasse ($s=3,9$), og standardavviket synker ($d=1,0$). Det tyder på at en gruppe studenter opplever en stor reduksjon i mestringsforventning i løpet av det første studieåret, mens i løpet av det andre studieåret jevner mestringsforventningen blant studentene seg ut igjen.

Frisvarspørsmålene peker i samme retning, siden studentene i 1. og 3. klasse stort sett sier at programmeringsoppgavene går greit, mens studentene i 2. klasse i større grad omtaler oppgavene som vanskelige og etterlyser mer opplæring. En andreårsstudent skriver for eksempel: «De [programmeringsoppgavene i fysikk] funker greit men de er på et høyere kompetansenivå innen programmering enn det vi har lært i intro kurset (ITGK). Noe som gjør

de ofte veldig krevende å gjennomføre øvingene og personlig har det gjort at jeg ikke ønsker å jobbe med det selv om jeg synes ITGK var veldig gøy fag.»

Den andre påstanden, som dreier seg om sammenheng mellom å like en oppgave og å klare å løse den, øker jevnt fra 1. til 3. klasse, men har omtrent samme, høye score i hvert trinn (fra lavest $s=3,8$ til høyest $s=4,0$). Det peker mot at det er en utbredt kobling mellom studentenes evne til å løse beregningsorienterte oppgaver og studentenes interesse for oppgavene. En tredjeårsstudent peker mot denne koblingen ved å påpeke forskjellen mellom hvordan det var i 1. klasse og hvordan det er i 3. klasse: «I 1. klasse opplevde jeg at de [programmeringsoppgavene] var for vanskelige, siden vi ikke hadde lært oss nok grunnleggende programmering. Nå synes jeg programmeringsøvingene er interessante, og jeg liker dem.»

Interesseverdi



Figur 6: Svar på påstandene fra konstruktet interesseverdi.

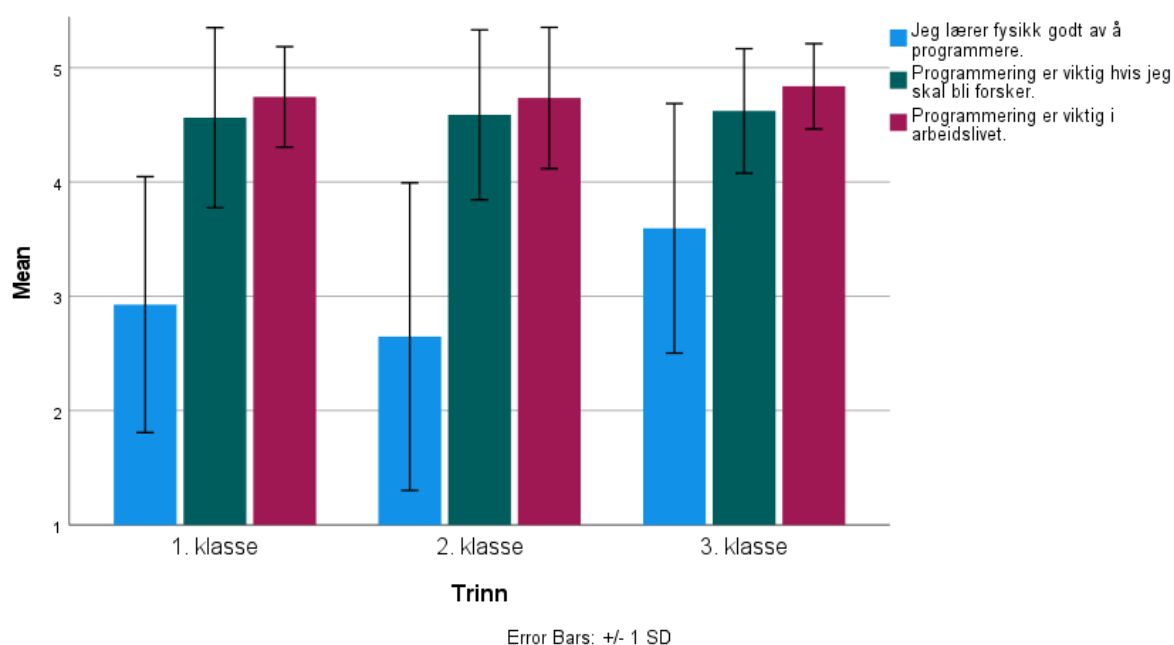
Figur 6 viser påstandene fra konstruktet interesseverdi: «Jeg er interessert i programmering generelt.» (blå søyle) og «Jeg er interessert i programmering i fysikkemner.» (grønn søyle).

Studentene opplever at beregningsorientert fysikk har ganske høy interesseverdi som ikke endrer seg stort i løpet av årene (fra lavest $s=3,9$ til høyest $s=4,2$). 1. og 2. klasse har litt høyere score for den generelle programmeringsinteressen ($s=4,2$) enn programmeringsinteressen i fysikkemner ($s=3,9$), mens i 3. klasse er interessen litt høyere for

programmering i fysikkemner ($s=4,2$) enn i programmering generelt ($s=4,0$). Man ser også et lavere standardavvik i blå søyle, første årstrinn, og grønn søyle, tredje årstrinn ($d=0.8$), enn i øvrige søyler ($d=1,0$). Dette tyder på at fysikkstudiene gjør at studentene får øynene mer opp for programmering i fysikk spesifikt innen de har nådd tredje klasse.

Fra frisvars spørsmålene nevner som regel studentene at programmering i fysikk er interessant når det bidrar til økt forståelse for fysikkstoffet, for eksempel ved å visualisere problemer. En førsteårsstudent skriver «Eg synest programmering gjev ein god intuisjon av korleis system oppfører seg. Det skaper også eit anna perspektiv på stoffet, som eg synest fører til betre forståing.»

Nytteverdi



Figur 7: Rød og grønn søyle tilsvarer påstander fra konstruktet nytteverdi, mens den blå søylen viser påstanden som er kopiert fra IMPEL-undersøkelsen.

Figur 7 viser påstandene fra konstruktet nytteverdi: «Programmering er viktig hvis jeg skal bli forsker.» (grønn søyle) og «Programmering er viktig i arbeidslivet.» (rød søyle), samt påstanden som er hentet fra IMPEL-prosjektet: «Jeg lærer fysikk godt av å programmere.» (blå søyle).

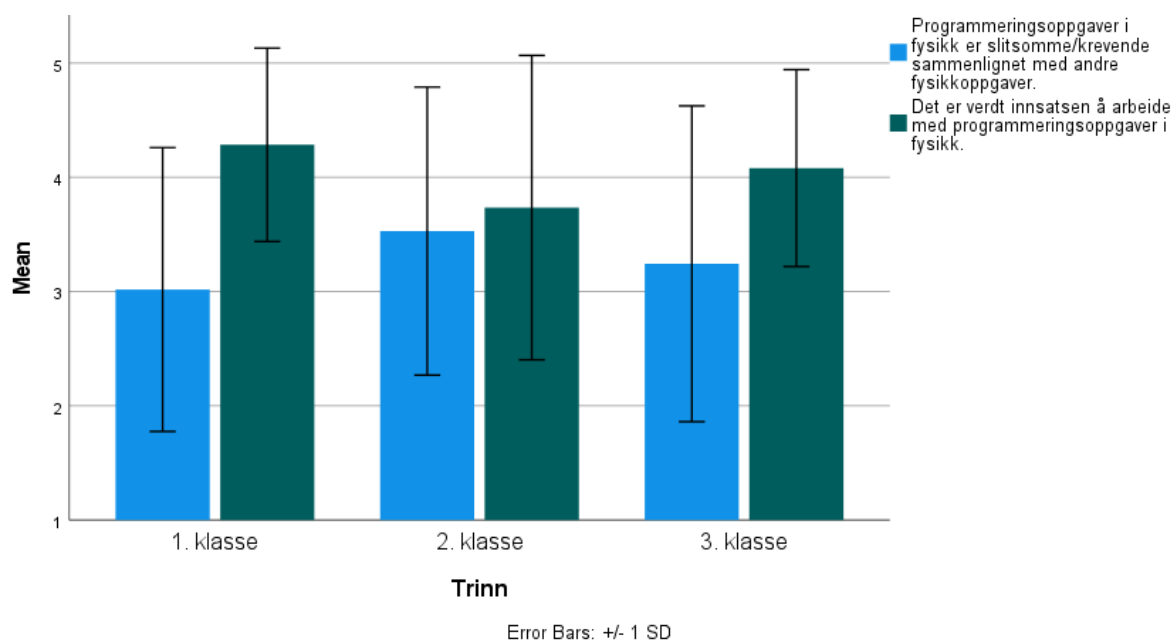
Grønn og rød søyle scorer høyt og likt blant alle klasser (alle verdier mellom $s=4,5$ og $s=4,8$) og standardavviket er jevnt og lavt (fra $d=0,7$ til $d=0,3$), spesielt lavt er standardavviket i 3. klasse. Studentene virker å være enige om at programmering er viktig for dem i fremtiden.

Enkelte studenter nevner også i frisvarspørsmålene at programmering er noe de ser på som viktig i fremtiden, en student i 1. klasse sier «Nivået er passe og jeg tror det er bra at vi bruker programmering i fysikk, siden dette er noe vi vil trenge fremover».

IMPEL-påstanden ligger nært midten ($s=2,9$, $d=1,1$) i 1. klasse, litt lavere ($s=2,6$, $d=1,3$) i 2. klasse og høyest i 3. klasse ($s=3,5$, $d=1,1$). Standardavviket er høyt for alle trinn, men spesielt høyt i 2. klasse. De to mest markante poengene å ta med seg fra denne påstanden er at det er stor uenighet blant studentene og at studentene i 3. klasse scorer klart høyest.

Det er delte meninger i frisvarspørsmålene om programmering gir godt læringsutbytte i fysikk, noen studenter mener det hjelper med å forstå fysikken, andre mener at programmeringsoppgavene havner på siden og virker lite relevant for fysikkpensum ellers. En tredjeårsstudent skriver: «Varierende, større prosjekter som det blir lagt opp for å bruke mye tid og med god veiledning har vært nyttige. Mindre, obligatoriske, oppgaver har jeg opplevd som unyttige i forhold til resten av øvingsopplegget. Jeg har ofte irritert meg over at jeg ikke kan jobbe med å øke forståelsen av faget i stedet.»

Kostnad



Figur 8: Svar på påstandene fra konstruktet kostnad.

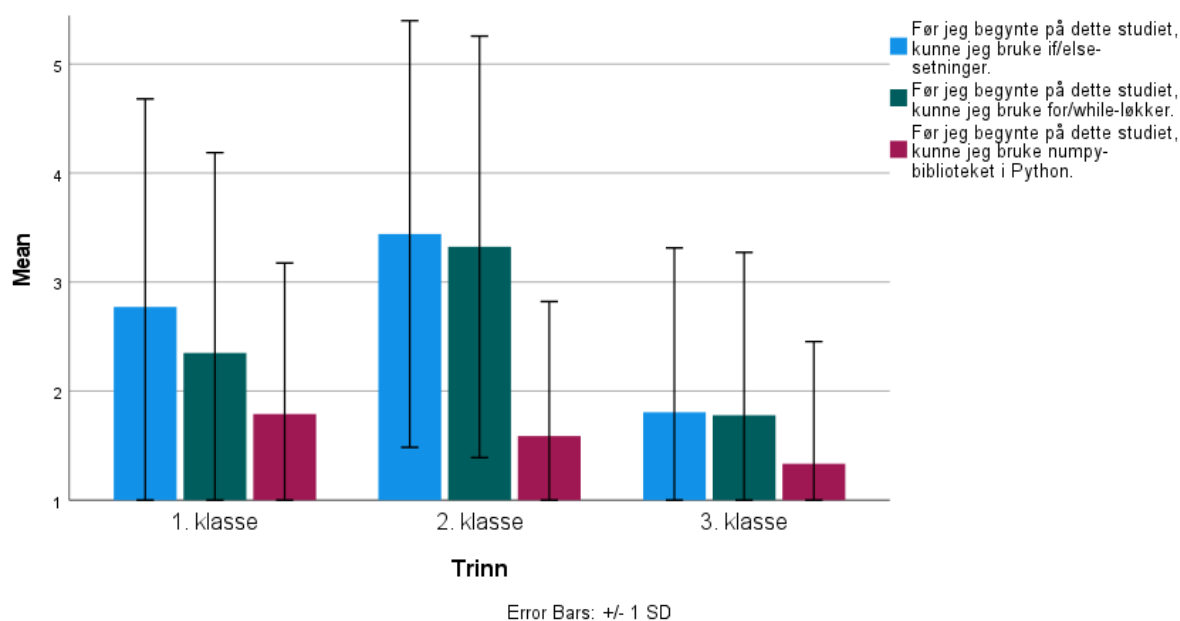
Figur 8 viser påstandene fra konstruktet kostnad: «Programmeringsoppgaver i fysikk er slitsomme/krevende sammenlignet med andre fysikkoppgaver.» (blå søyle) og «Det er verdt innsatsen å arbeide med programmeringsoppgaver i fysikk.» (grønn søyle).

Blå søyle i figur 8 ligger ganske stabilt nær midten (fra $s=3,0$ til $s=3,5$ til $s=3,2$) med et jevnt, høyt standardavvik (fra $d=1,2$ på det minste til $d=1,3$ på det meste). Studentene uttrykker at i gjennomsnitt at programmeringsoppgaver er omtrent like krevende som andre fysikkoppgaver, men det er stor variasjon blant studentene.

En andreårsstudent uttrykker i frisvarspørsmålene: «Bruker ofte lang tid med mye frustrasjon underveis, men til slutt, når oppgavene er løst, sitter jeg igjen med mye kunnskap og en mestringsfølelse.» Dette peker mot at det er en kobling mellom at oppgave tar mye tid og at de er frustrerende å jobbe med.

Grønn søyle scorer ganske høyt og har ganske lik score i 1. og 3. klasse ($s=4,2$ og $s=4,0$) med lignende standardavvik ($d=0,8$ for begge), og litt lavere score i 2. klasse ($s=3,7$) med høyere standardavvik ($d=1,3$). Her ser det ut til at en gruppe studenter synes det er mindre verdt innsatsen å jobbe med programmeringsoppgaver i fysikk i 2. klasse, men denne gruppen synes ikke i resultatene fra 3. klasse.

Forhåndskunnskaper



Figur 9: Svar på påstandene fra konstruktet forhåndskunnskaper.

Figur 9 viser påstandene fra konstruktet forhåndskunnskaper: «Før jeg begynte på dette studiet, kunne jeg bruke if/else-setninger.» (blå søyle), «Før jeg begynte på dette studiet, kunne jeg bruke for/while-løkker.» (grønn søyle) og «Før jeg begynte på dette studiet, kunne jeg bruke numpy-biblioteket i Python.» (rød søyle).

Blå og grønn søyle følger hverandre jevnt og ligger litt under midten i 1. klasse (blå: $s=2,8$, grønn: $s=2,3$, $d=2,0$ for begge), litt over middelværdien i 2. klasse (blå: $s=3,3$, grønn: $s=3,2$, $d=2,0$ i begge) og lavt i 3. klasse ($s=1,8$ i begge, $d=1,4$ i begge). Det høye standardavviket kan forklares med at de fleste studenter svarte enten 1 eller 5. Rød søyle scorer lavere enn 2 i alle årstrinn. Det er noe overraskende at 2. klasse scorer høyere enn 1. klasse, det forventede ville enten være at de scorer ganske like eller at 1. klasse scorer høyere enn 2. klasse, siden mengden programmering i grunnopplæringen er økende.

Korrelasjon for hele utvalget

Videre kommer fire matriser som inneholder Cronbachs alfa (alfa) og Pearson-korrelasjon (p) mellom påstander og konstrukter. Ruter med 0 har ikke signifikant korrelasjon, ruter med andre tall viser til signifikant korrelasjon mellom de to variablene. Der en påstand inngår i konstruktet den sammenlignes med er feltet markert med en strek, og korrelasjonen er derfor ikke inkludert. Jeg tar først for meg korrelasjonen i hele utvalget.

Tabell 8: Korrelasjonsmatrise der hele utvalget ($N=128$) er inkludert.

Påstand\Konstrukt	Mestringsforventning	Interesseverdi	Nytteverdi	Kostnad	Forhåndskunnskaper
Cronbachs alfa	0.453	0.777	0.687	0.487	0.875
2.3 Jeg forventer at jeg klarer å løse programmeringsoppgaver i fysikk.	-	0.406	0	-0.494	0.264
2.4 Hvorvidt jeg liker en programmeringsoppgave avhenger av i hvilken grad jeg klarer å løse den. (Reversert)	-	0.205	0	-0.360	0.213
2.1 Jeg er interessert i programmering generelt.	0,299	-	0.185	-0.443	0.363
2.2 Jeg er interessert i programmering i fysikkemner.	0.394	-	0.189	-0.566	0.276
2.6 Programmering er viktig hvis jeg skal bli forsker.	0	0	-	0	0
2.7 Programmering er viktig i arbeidslivet.	0	0.201	-	0	0.191
2.11 Programmeringsoppgaver i fysikk er slitsomme/krevende sammenlignet med andre fysikkoppgaver.	-0.492	-0.479	0	-	-0.274
2.12 Det er verdt innsatsen å arbeide med programmeringsoppgaver i fysikk. (Reversert)	-0.378	-0.433	-0.197	-	0
2.8 Før jeg begynte på dette studiet, kunne jeg bruke if/else-setninger.	0.230	0.323	0	0	-
2.9 Før jeg begynte på dette studiet, kunne jeg bruke for/while-løkker.	0.233	0.296	0.202	0	-
2.10 Før jeg begynte på dette studiet, kunne jeg bruke numpy-biblioteket i Python.	0.377	0.346	0	-0.334	-
2.5 Jeg lærer fysikk godt av å programmere.	0.243	0.420	0.320	-0.545	0.190

Mestringsforventning (påstand 2.3) korrelerer sterkt med interesseverdi ($p=0,406$) og kostnad ($p=-0,494$). Det er også en svakere korrelasjon med forhåndskunnskaper ($p=0,264$).

Konstruktet interesseverdi korrelerer med påstand 2.7 ($p=0,201$), men ikke med påstand 2.6. Ellers er det en tydelig korrelasjon mellom interesseverdi og kostnad ($p=-0,479$) og mellom interesseverdi og forhåndskunnskaper ($p=0,323$, $p=0,296$, $p=0,346$). Interesseverdi korrelerer med IMPEL-påstanden ($p=0,420$).

Konstruktet nytteverdi har ingen korrelasjon med studentenes mestringsforventning. Nytteverdi korrelerer med påstandene i interesseverdi ($p=0,185$, $p=0,189$). Det er ingen korrelasjon mellom nytteverdi og kostnad og det er varierende korrelasjon med forhåndskunnskaper ($p=0$ for 2.8 og 2.10 og $p=0,202$ for 2.9). Nytteverdi korrelerer mest med IMPEL-påstanden ($p=0,320$).

Kostnad (påstand 2.11) korrelerer med alt unntatt nytteverdi.

Konstruktet forhåndskunnskaper har jevnt over en svak korrelasjon med de andre konstruktene, med unntak av påstand 2.6 og påstand 2.12, hvor det ikke er korrelasjon.

Korrelasjon for studenter i første årstrinn

Over til studentene i første årstrinn. Her støtter ikke cronbachs alfa at påstandene i nytteverdi utgjør den samme dimensjonen ($\alpha=0,453$).

Tabell 9: Korrelasjonsmatrise med studenter i første årstrinn (N=57).

Påstand\Konstrukt	Mestringsforventning	Interesseverdi	Nytteverdi	Kostnad	Forhåndskunnskaper
Cronbachs alfa	0.280	0.810	0.453	0.528	0.895
2.3 Jeg forventer at jeg klarer å løse programmeringsoppgaver i fysikk.	-	0.372	0	-0.334	0.505
2.4 Hvorvidt jeg liker en programmeringsoppgave avhenger av i hvilken grad jeg klarer å løse den. (Reversert)	-	0	0	-0.423	0
2.1 Jeg er interessert i programmering generelt.	0.270	-	0	-0.435	0.519
2.2 Jeg er interessert i programmering i fysikkemner.	0.345	-	0	-0.646	0.444
2.6 Programmering er viktig hvis jeg skal bli forsker.	0	0	-	0	0
2.7 Programmering er viktig i arbeidslivet.	0	0	-	0	0
2.11 Programmeringsoppgaver i fysikk er slitsomme/krevende sammenlignet med andre fysikkoppgaver.	-0.351	-0.465	0	-	-0.363
2.12 Det er verdt innsatsen å arbeide med programmeringsoppgaver i fysikk. (Reversert)	-0.540	-0.556	0	-	-0.312
2.8 Før jeg begynte på dette studiet, kunne jeg bruke if/else-setninger.	0.319	0.536	0	-0.387	-
2.9 Før jeg begynte på dette studiet, kunne jeg bruke for/while-løkker.	0.392	0.463	0	-0.340	-
2.10 Før jeg begynte på dette studiet, kunne jeg bruke numpy-biblioteket i Python.	0.392	0.397	0	-0.398	-
2.5 Jeg lærer fysikk godt av å programmere.	0.323	0.561	0	-0.615	0.387

Mestringsforventning (2.3), interesseverdi, kostnad og forhåndskunnskaper korrelerer med hverandre, mens nytteverdi (2.6 og 2.7) har ingen korrelasjon. Påstand 2.4 korrelerer kun med kostnad ($p=-0.423$). Interesseverdi og forhåndskunnskaper korrelerer med påstand 2.5 ($p=0,561$ og $p=0,387$).

Korrelasjon for studenter i andre årstrinn

For andreårsstudentene gir konstruktanalysen samme resultater som for det totale studentutvalget, med høye nok alfaverdier for interesseverdi (alfa=0,748), nytteverdi (alfa=0,800) og forhåndskunnskaper (alfa=0,828). Sammenlignet med tidligere korrelasjonsmatriser er det langt mindre korrelasjon mellom konstruktene i dette tilfellet.

Tabell 10: Korrelasjonsmatrise for studenter i andre årstrinn (N=34).

Påstand\Konstrukt	Mestringsforventning	Interesseverdi	Nytteverdi	Kostnad	Forhåndskunnskaper
Cronbachs alfa	0.524	0.748	0.800	0.514	0.828
2.3 Jeg forventer at jeg klarer å løse programmeringsoppgaver i fysikk.	-	0.461	0	-0.548	0
2.4 Hvorvidt jeg liker en programmeringsoppgave avhenger av i hvilken grad jeg klarer å løse den. (Reversert)	-	0	0	-0.469	0
2.1 Jeg er interessert i programmering generelt.	0	-	0	-0.424	0
2.2 Jeg er interessert i programmering i fysikkemner.	0.526	-	0	-0.590	0
2.6 Programmering er viktig hvis jeg skal bli forsker.	0	0	-	0	0
2.7 Programmering er viktig i arbeidslivet.	0	0	-	0	0.383
2.11 Programmeringsoppgaver i fysikk er slitsomme/krevende sammenlignet med andre fysikkoppgaver.	-0.548	-0.430	0	-	0
2.12 Det er verdt innsatsen å arbeide med programmeringsoppgaver i fysikk. (Reversert)	-0.468	-0.500	0	-	0
2.8 Før jeg begynte på dette studiet, kunne jeg bruke if/else-setninger.	0	0	0	0	-
2.9 Før jeg begynte på dette studiet, kunne jeg bruke for/while-løkker.	0	0	0	0	-
2.10 Før jeg begynte på dette studiet, kunne jeg bruke numpy-biblioteket i Python.	0	0	0	-0.345	-
2.5 Jeg lærer fysikk godt av å programmere.	0	0	0.341	-0.439	0

Mestringsforventning (2.3) korrelerer med interesseverdi ($p=0,461$) og kostnad ($p=-0,548$), mens det er ingen signifikant korrelasjon med nytteverdi eller forhåndskunnskaper.

Det er korrelasjon mellom interesse og kostnad (2.11) ($p=-0,430$). I motsetning til hos førsteårsstudentene, korrelerer ikke interesseverdi med forhåndskunnskaper.

Nytteverdikonstruktet har svært lite korrelasjon med andre konstrukter, den eneste signifikante korrelasjonen er med påstand 2.5 (0,341).

Forhåndskunnskaper spiller svært lite inn på andreårsstudentenes motivasjon, det er kun korrelasjon mellom forhåndskunnskaper og påstand 2.7 ($p=0,383$).

Korrelasjon for studenter i tredje årstrinn

For tredjeårsstudentene er igjen konstruktanalysen lik som for det totale utvalget ($\alpha > 0,7$ for interesseverdi, nytteverdi og forhåndskunnskaper).

Tabell 11: Korrelasjonsmatrise for studenter i tredje årstrinn (N=37).

Påstand\Konstrukt	Mestringsforventning	Interesseverdi	Nytteverdi	Kostnad	Forhåndskunnskaper
Cronbachs alfa	0.579	0.748	0.751	0.278	0.909
2.3 Jeg forventer at jeg klarer å løse programmeringsoppgaver i fysikk.	-	0.479	0.385	-0.523	0
2.4 Hvorvidt jeg liker en programmeringsoppgave avhenger av i hvilken grad jeg klarer å løse den. (Reversert)	-	0	0	0	0.528
2.1 Jeg er interessert i programmering generelt.	0.370	-	0	-0.543	0.327
2.2 Jeg er interessert i programmering i fysikkemner.	0.438	-	0.388	-0.504	0
2.6 Programmering er viktig hvis jeg skal bli forsker.	0.338	0	-	-0.340	0
2.7 Programmering er viktig i arbeidslivet.	0.341	0.403	-	-0.467	0
2.11 Programmeringsoppgaver i fysikk er slitsomme/krevende sammenlignet med andre fysikkoppgaver.	-0.567	-0.571	0	-	-0.333
2.12 Det er verdt innsatsen å arbeide med programmeringsoppgaver i fysikk. (Reversert)	0	0	-0.409	-	0
2.8 Før jeg begynte på dette studiet, kunne jeg bruke if/else-setninger.	0.408	0	0	0	-
2.9 Før jeg begynte på dette studiet, kunne jeg bruke for/while-løkker.	0.404	0	0	0	-
2.10 Før jeg begynte på dette studiet, kunne jeg bruke numpy-biblioteket i Python.	0.443	0	0	0	-
2.5 Jeg lærer fysikk godt av å programmere.	0	0.460	0.453	-0.660	0

Mestringsforventning (2.3) har korrelasjon med interesseverdi ($p=0,479$), nytteverdi ($p=0,385$) og kostnad ($p=-0,523$), men ikke forhåndskunnskaper.

Interesseverdi har en delvis korrelasjon med nytteverdi, siden interesseverdikonstruktet korrelerer med påstand 2.7 ($p=0,403$), men ikke påstand 2,6, og nytteverdikonstruktet korrelerer med påstand 2.2 ($p=0,388$), men ikke påstand 2.1.

Nytteverdi korrelerer med påstand 2.12 ($-0,409$).

Kostnad (2.11) korrelerer med interesseverdi ($p=-0,571$) og forhåndskunnskaper ($p=-0,333$), men ikke med nytteverdi.

Forhåndskunnskaper korrelerer med påstand 2.4 ($p=0,528$), påstand 2.1 ($p=0,327$) og påstand 2.11 ($p=-0,333$)

Påstander rettet mot hvert enkelt emne

Videre følger beskrivelser av søylediagrammene i vedlegg F med påstandene fra del 3 i spørreskjemaet der fire påstander ble evaluert for hvert enkelt fysikkemne studentene hadde hatt. Hvert vedlegg viser to søylediagrammer; ett for 2. klasse og ett for 3.klasse. Hver søyle tilsvarer ett fysikkemne: Mekanisk fysikk (blå søyle), Elektrisitet og magnetisme (grønn søyle), Bølgefysikk og fluidmekanikk (rød søyle), Termisk fysikk (oransje søyle), Innføring i kvantefysikk (brun søyle) og Optikk (lysegrønn søyle). Siden det er tre fysikkemner per årstrinn, er de tre søylene til venstre førsteårsemner og de tre søylene til høyre er andreårsemner. Se tabell 1 for oversikt over alle emner i 1. og 2. klasse.

Jeg hadde fått god nok opplæring i programmering til å løse programmeringsoppgavene. Vedlegg F1 viser at 2. og 3. klasse svarer ganske likt angående fysikkemnene de hadde i første årstrinn, med jevnt høyere score i 2. klasse. Elektrisitet og magnetisme får lavere score enn øvrige førsteårsemnene. Når det gjelder fysikkemnene som kun tredjeårsstudentene har hatt, altså andreårsemnene, får de en jevnt over høyere score enn førsteårsemnene. 3. klasse gir andreårsemnene en høyere score enn førsteårsemnene, spesielt Optikk får høy score.

Programmeringsøvingene krevde numerikkferdigheter jeg ikke hadde lært.

Vedlegg F2 viser at 3. klasse gir andreårsemnene dårligere score enn førsteårsemnene. Førsteårsemnene får generelt en høyere score i favør programmering fra 2. klasse enn fra 3. klasse, som kan ha et par ulike årsaker. En mulig årsak er at 3. klasse sammenligner førsteårsemnene med andreårsemnene og de gir generelt andreårsemnene en høyere score i favør programmering, som kan medføre at de i ettertid ser førsteårsemnene i et mer negativt lys. En annen mulig årsak er at studentene har et behov for å gi et positivt inntrykk av deres nåværende situasjon, som fører til en tendens til å svare mer positivt jo nærmere nåtiden påstanden gjelder.

Jeg foretrakk regneøvinger fremfor programmeringsøvinger.

I vedlegg F3 får alle emnene en ganske jevn score i begge årstrinn (s varierer mellom 3,8 og 3,2). I 3. klasse er det en jevn, synkende trend fra 1. til 4. semester. I alle fysikkemner uttrykker studentene i gjennomsnitt en preferanse for regneøvinger. 3. klasse gir generelt høyere score for førsteårsemnene enn 2. klasse i favør regneøvinger.

Jeg mestret programmeringsøvingene bedre enn regneøvingene.

I vedlegg F4 gir 3. klasse andreårselevene høyere score enn førsteårselevene. Optikk er det eneste emnet der studentene mener de mestret programmeringsøvingene bedre enn eller like godt som regneøvingene. Mekanisk fysikk og Elektrisitet og magnetisme får lavere score enn øvrige emner. 2. klasse gir førsteårselevene høyere score enn 3. klasse, med en jevn differanse på omtrent 0,5.

Frisvarspørsmål

I spørreskjemaet var det to frisvarspørsmål, svarene på dem har jeg analysert tematisk i NVivo. Det er også gjort betraktninger for hvordan studentene svarer i ulike årstrinn.

Mange studenter opplever at programmering i fysikk bidrar til økt forståelse for fysikk, spesielt de mer abstrakte grenene av fysikk. Et ord som gjentas ofte er «praktisering», altså at man får gjort fysikk mer praktisk. Andre studenter uttrykker det motsatte synspunktet, nemlig at programmeringsoppgavene føles løsrevet fra fysikken og at det er lite relevant for emnet som helhet.

Mange studenter, spesielt i andre årstrinn, synes vanskelighetsgraden er for høy og beskriver programmeringsoppgavene i fysikk som vanskelige, krevende og omfattende. Noen bruker ord som overveldende og frustrerende. I analysen kom det fram at de ofte ikke har fått så mye opplæring i forkant av programmeringsoppgavene og at det er mye man må «skjønne selv». Noen ser på dette som en god utfordring, andre ser på det som et problem som har gjort at de ofte ikke får det til.

Mange studenter nevner grunnkurset i programmering (ITGK) som studentene har i første semester og sier at nivået i på programmeringen fysikkemnene er høyere enn det de lærer i ITGK. De mener ITGK ikke forbereder en nok til programmering i fysikk, spesielt med tanke på biblioteker i Python som brukes mye i fysikk, men som nevnes lite i ITGK. Et annet emne som nevnes flere ganger av tredjeårsstudentene er fjerdesemesteremnet Introduksjon til vitenskapelige beregninger (vitber). Studentene skriver at problemene de hadde med det programmeringstekniske ofte løste seg i forbindelse med dette emnet, etter det var ikke programmeringen vanskelig lenger.

Spesielt blant tredjeårsstudentene er det flere som gir en beskrivelse av hvordan deres forhold til programmering har utviklet seg i løpet av studiet. Den gjennomgående

beskrivelsen er at programmering var svært krevende i første og delvis andre studieår, mens etter det så har programmeringen vært morsom og interessant.

5.2 Intervju

Denne seksjonen tar for seg temaene og undertemaene fra den tematiske analysen av intervjuene (vedlegg E og tabell 7) ved hjelp av sitater og forklarende/utdypende tekst. De tre intervjuene var individuelle og deltakerne omtales i rapporten med de anonyme kodenavnene K2, M2 og K3, der K/M står for kvinne/mann og tallet bak referer til hvilket studietrinn studenten gikk på. Alle de tre intervjudeltakerne gikk på studieprogrammet MTFYMA (Fysikk og matematikk sivilingeniør).

Forhåndskunnskaper og -forventninger

Spørreundersøkelsen viste at det er stor variasjon i studentenes programmeringsferdigheter når de begynner å studere på NTNU. De tre intervjuinterdeltakerne ble også bedt om å fortelle hvor mye de hadde programmert i forkant av studiene, og i to av intervjuene kom vi inn på temaet om hva slags forventninger de hadde til programmeringsopplæringen i forkant av studiene. K2 hadde hatt et programmeringsemne i videregående skole i samarbeid med NTNU, M2 kunne allerede mye programmering (omtrent tilsvarende pensum i ITGK) siden faren hans jobbet med IT, og K3 hadde aldri programmert før. Alle tre gir inntrykk av at de forventet at programmering ville inngå i fysikkstudiene, men K3 understreker at hun verken var forberedt på hvor stor del av pensum det skulle være eller hvor nyttig det var.

Motivasjon

Videre følger de fire undertemaene som inngår i motivasjonsbegrepet slik det er definert i EV-modellen, altså er motivasjon delt inn i to hovedkomponenter: mestringsforventning og subjektiv verdi. Den subjektive verdien er her delt inn i interesseverdi, nytteverdi og kostnad.

Mestringsforventning

Studentenes motivasjon er blant annet påvirket av mestringsforventning, som er et av undertemaene fra den tematiske analysen. Studentene beskriver mestringsfølelse som en viktig motivasjonskilde, nemlig å få koden til å fungere. Studenten K3 går så langt som til å si at mestringsfølelse er mye av det som gjør at studentene orker å gå gjennom studiet. For å få innsyn i studentenes mestringsforventning og mestringsfølelse, ble det spurt om studentene opplever at de får til oppgavene og hvilken vanskelighetsgrad

programmeringsoppgavene har. Andreårsstudentene K2 og M2 er enige om at programmeringsoppgaver vanligvis er vanskeligere enn de vanlige regneoppgavene.

K2: *Jeg synes som oftest de er hakket vanskeligere, det er mye mer forventet av deg når du skal gjøre numerisk, og det er gjerne litt sånn mere store oppgaver, gjerne litt mere komplekse oppgaver, som vi får som numeriske oppgaver. Så er det jo ofte hakket vanskeligere, på en måte konsepter som vi ikke har vært så mye inne på før, at det er sånn «oi, nå tar vi for oss et helt tema, bare med dette».*

K2 sin beskrivelse tyder på at programmeringsoppgavene går i dybden på fagområder som studentene i forkant av oppgavene ikke forventet skulle være så store eller viktige. Dette kan gå ut over studentenes motivasjon for å jobbe med oppgavene.

M2: *Mens de programmeringsoppgavene vi hadde fått, de var... de var litt vanskelige, de var litt krevende, men de var ikke sånn umulige, på en måte. Det var ikke sånn at jeg så ikke hvor jeg skulle begynne, det var aldri sånn... jeg kunne se litt hvor jeg skulle begynne også gjennom å snakke litt mer og å tenke litt så til slutt så kom jeg fram til et eller annet som... ble rett. Så er det mange fysikkoppgaver der jeg av og til bare satt fast og bare visste ikke hvor jeg skulle begynne engang.*

M2 påpeker at programmeringsoppgavene ikke har så stor variasjon i vanskelighetsgrad, men holder seg på et jevnt, høyt nivå. Han understreker dog at han har god erfaring med å klare å løse programmeringsoppgavene, selv om de er vanskelige. K2 er enig i at vanskelighetsgraden i programmering er høy og lite variert.

K2: *Jeg føler kanskje de vanlige er mer sånn her bygger vi de opp også er det sånn her har du kanskje en litt lettere oppgave, også får du litt vanskeligere og vanskeligere, også får du kanskje fasit etterpå. Men programmeringsoppgavene er sånn du får ikke noe sånn jevnlig oppgaver, det er sånn her har du en stor oppgave, vær så god.*

Tredjeårsstudenten K3 anser programmeringsoppgavene totalt sett for å være omtrent like vanskelige å løse som de vanlige regneoppgavene. Oppgavene er i seg vanskeligere, men bruken av datamaskiner gjør det enklere, siden maskinen kan gjøre mye av arbeidet for deg. Dette kan peke på at K3 opplever programmeringsoppgaver i fysikk på en annen måte enn K2 og M2 siden hun er mer komfortabel eller vant med å bruke programmering som et verktøy i fysikk.

K3 nevner også at alle problemløsningsoppgaver, enten det er innen programmering eller analytisk fysikk, oppfattes som overveldende, som innebærer at det ligger i oppgavenes natur at man ikke kan forvente at man vil klare å løse dem.

K3: *Fordi problemløsning handler jo på en måte om at du bare, det handler jo ikke så mye om innholdet i faget, men bare at uansett hvilket tema det er, måten du liksom går fram da, og det er jo det på en måte de sier problemløsning er, ikke, på en måte, selve beregningen i seg selv, men det derre... at du bare får en litt sånn overveldende... et overveldende problem da, også er det det å liksom... ehm dele det opp, og på en måte, lage en plan i hodet, som jeg tror man gjør litt underbevisst da, bare at vi har gjort det så mye nå, så nå bare begynner vi med en gang. Men at det som skjer i underbevisstheten er at du på en måte tar opp problemet, analyserer det, eh finner ut av som egentlig det ble spurt om, og... å lage en liksom plan for hvordan du vil gå frem og sånn. Og jeg tror at programmering er med å på en måte utvikle den egenskapen da.*

På denne måten tillegges programmeringen ekstra nytteverdi siden den gir generelle problemløsningsferdigheter.

K2 understreker at hennes mestring i programmering er tett koblet til muligheten studentene får til å arbeide med programmering jevnlig.

Meg: *Hva var det som var annerledes i mekanisk fysikk, i forhold til de fysikkfagene du har nå?*

K2: *Det er vel det at avstanden fra ITGK er kortere.*

Meg: *Til det man gjør i programmering der.*

K2: *Ja. Og både da på en måte har man programmering jevnlig, så man kan på en måte, da er det liksom som om programmeringen ikke er så skummel, for da vet man på en måte at det gjør vi jevnlig, det gir mening. Også er det sånn «åja nå har jeg ikke programmert på et halvt år nå, åssen gjør man det her igjen?».*

I semestrene der K2 får jobbe med programmering jevnlig i rene programmeringsemner, har hun også høyere mestringsforventning når hun programmerer i fysikk. Strukturen i fysikkemnene er at det er et lite antall programmeringsøvinger per semester som hver for

seg er relativt omfattende, derfor utvikler ikke studentene noen fast vane med å programmere. For K3 er dette annerledes siden hun har hatt en sommerjobb hvor hun brukte mesteparten av tiden på å programmere, slik at hun fikk innarbeidet gode programmeringsvaner jevnlig over tid.

De tre studentene er enige om at det går ut over mestringsforventningen når de opplever at beregningsorienterte oppgavene både er krevende med tanke på fysikken og med tanke på programmeringen.

Meg: *Hva vil du si var det som gjorde det vanskelig for deg å tenke rett, på en måte?*

M2: *Ehm... hmm... det var mye av det at... kanskje det å forstå... og da måtte man på en måte forstå fysikken og programmering. Hvis jeg for eksempel ikke var helt sikker på hva fysikken bak det var, og ikke helt var sikker på funksjonene jeg brukte i programmeringen. Og... så det var liksom... normalt i fysikken, hvis man holder på med oppgaver, så er det en ting man kanskje ikke helt forsto, som gjør at man gjør det feil. Også når man til slutt forstår den tingen, så får man oppgaven rett. Men nå var det liksom både fysikkforståelsen som var litt sånn... vi hadde kanskje ikke fulgt godt nok med eller ja, på akkurat det de tok opp. I tillegg til at programmeringsforståelsen min ikke var helt... var der ennå, på en måte. Da var det liksom mye som var feil, som gjorde at oppgaven var vanskelig, så jeg visste ikke om det var programmeringen eller fysikkforståelsen som var det som var feil da.*

Når både fysikken og programmeringen oppleves som vanskelig, sliter studentene med å få programmet sitt til å fungere siden de ikke vet hva slags feil de har. M2 forklarer videre at han vanligvis får rettet opp i feilene i koden sin ved å teste om det er noe feil med det programmeringstekniske, og hvis det ikke er tilfellet, så kan han anta at det dreier seg om fysikken.

Studentene uttrykker at det er en sammenheng mellom mestring og om de opplever at kravene de møter fra universitet er tydelige og rimelige.

Meg: *I hvilken grad synes du at kravene i programmeringsoppgavene er rimelige?*

K2: *Jeg vet egentlig ikke helt hva som er kravene i programmeringsoppgavene.*

Meg: *Hva synes du om det?*

K2: *Ehh jeg vet ikke, men jeg føler at det er veldig... dramatisert. Hvis du ikke får til disse oppgavene, så kan du ikke gå opp til eksamen. Men det er kanskje litt sånn... i forhold til hvor lite de underviser i programmering. Vi har jo hatt programmeringseksempler nå i kvante... introduksjon til kvante, men det er første... nesten første gang vi får programmeringseksempler i forelesningene. Og da er det sånn de underviser ikke i det i forelesninger, men de forventer at vi skal kunne det. Så det synes jeg kanskje er litt... ja, urimelig. Det er sånn, hadde de kanskje nevnt det innimellom, så hadde det gitt mer mening at vi skal kunne det. Men når man vanligvis ikke egentlig ser noe til det, så er det sånn «åja dere forventer at vi skal kunne dette, men vi har ikke lært noe om det egentlig.»*

Her trekker K2 fram at det virker urimelig at programmeringsoppgavene er nødvendige for å få lov til å gå opp til eksamen og få faget godkjent, samtidig som programmering i de fleste emner ikke inngår i undervisningen. Det tyder på at det er en forskjell mellom hva det forventes av studentene og omfanget av læringsressurser og -aktiviteter for å realisere disse forventningene.

K3 har en lignende innfallsvinkel:

Meg: *Ehh i hvilken grad synes du at kravene i programmeringsoppgavene er rimelige? På en måte kravene som stilles til deg, for å kunne løse de?*

K3: *Jeg synes egentlig at det er... for å kunne løse det, eller sånn... fordi sånn generelt da, så synes jeg det er veldig vanskelig å vite hva de forventer i de numeriske... [...] Sånn, jeg, vi sitter ofte der også får vi et resultat da, også er det ikke helt riktig, også skjønner vi ikke om de på en måte forventer at vi skal få det eksakte resultatet eller om det er på en måte, ja, så jeg synes det er veldig vanskelig å vite hvilke krav som stilles, egentlig.*

Meg: *Ja. Hvordan oppleves det for deg?*

K3: *Det er jo det som er det mest frustrerende da, med... numerikken, på en måte. Jeg liker selve prosessen, men det er litt sånn, når man har resultatet, så vet du aldri, på en måte, om det er det de har villet da.*

K3 opplever at hun får et resultat, men har ingen måte å få bekreftet om det hun har gjort er rett. Dette kan gå ut over både mestringsfølelse og læringsutbytte.

Interesseverdi

Det neste undertemaet innen motivasjon dreier seg om interesseverdi, altså i hvilken grad studentene synes programmering er interessant og gøy. Intervjudeltakerne forteller om mange ulike aspekter de synes er interessant og gøy med programmering, derav blant annet å lage kode for å plote grafer og deretter se hvordan resultatet blir (spesielt hvis resultatet er riktig), å lage animasjoner (igjen spesifisert at det er morsomst hvis animasjonene er korrekte), å finne småting på nettet som man kan bruke (selv om det ikke er pensum) og å sette sammen mange små deler til et stort program. K3 sier at hun nå er svært interessert i programmering og at interessen har utviklet seg i løpet av studiet. K2 derimot har gradvis mistet interesse for programmering siden hun ikke lenger føler at hun får det til like godt som hun gjorde.

K2: *[...] jeg føler jo at jeg har jo mistet mye av lysten til å programmere fordi at jeg føler at jeg ikke får det til. Eh og da er det jo litt sånn, men jeg fikk jo til programmeringsoppgavene i ITGK, det var jo kjempelett. Jeg hadde... jeg syntes jo det var kjempegøy, så da er det på en måte hva er det som er grunnen til at jeg ikke lenger har lyst til å programmere, at jeg ikke ser på de øvingene engang. Selv om jeg føler at jeg syntes det var ganske greit da. For det var jo, når vi hadde mekanisk fysikk, så var det jo greit nok, for da var det «okei, jeg skjønner kanskje ikke fysikkdelen, men jeg kan programmeringsdelen, så kan noen andre hjelpe til med fysikkdelen». Så på en måte jobber vi sammen. Og det var jo ganske grei, på en måte, oppbygning. Men nå er det sånn «nei.».*

M2 har hatt en stabilt høy interesse for programmering gjennom studiet, selv om interessen for de vanlige regneoppgavene har blitt redusert.

Nytteverdi

Studentene er alle enige om at programmering er svært nyttig. De nevner en rekke ulike måter programmering kan være nyttig, som at man kan forstå fysikk bedre, man blir god på problemløsning, man lærer å tenke logisk, man får en mer realistisk tilnærming til fysikkproblemer, man har bruk for det i fremtidig jobb, man har stor frihet med tanke på

hvordan man vil løse oppgaven og datamaskiner kan regne mye fortere enn mennesker. Studentene nevner samtidig flere faktorer som reduserer nytteverdien.

M2: *På en måte, jeg... det er veldig greit at vi er flere personer til å løse oppgaven, jeg hadde jo ikke klart å løse den alene. Men, hvor mye jeg lærer ut av det, er jo egentlig en annen sak. Fordi at, jeg vil jo gjerne ha den gjort, og jeg vil jo gjerne få gått opp til eksamen så jeg får hatt den så jeg blir ferdig med faget, men ja.*

Som M2 uttrykker, kan fokuset fort bli å bare få programmeringsoppgavene unnagjort for å kunne gå opp til eksamen, siden programmering ikke inngår i selve eksamenspensumet i de fleste fysikkemner. Dermed tenker studentene mindre på hva de faktisk kan lære eller få ut av å løse oppgavene. K2 og K3 nevner også dette som en problemstilling de enten kjenner på selv eller ser hos medstudenter.

Det er vanskeligere å se nytten i programmeringsoppgavene for å forstå fysikk dersom programmeringen er for vanskelig.

K3: *Eh i første... første og andre semester da, på en måte i første klasse, da syntes jeg at vi ikke hadde godt nok grunnlag til å på en måte få noe utnytte av de numeriske øvingene. Eh at det handlet mer om å på en måte... at vi var ikke gode nok i programmering, jeg syntes at... At fysikken på en måte... eh, eller mens vi på en måte var på for-løkker og sånt i ITGK, for eksempel da, så husker jeg, så skulle vi drive å lage masse funksjoner og sånt, som egentlig var over vårt kompetansenivå da. Og det føler jeg egentlig gikk gjennom hele det første året, at... at det ble mye mer fokus på å bare få koden til kjøre og bare få til det oppgaven spør om, og ikke på en måte, ehh ja, hvordan skal jeg si det, ikke bry seg så mye om selve fysikken da.*

K3 påpeker at dette var et problem i starten av studiet, mens over tid har studentene blitt flinkere i å programmere og dermed fått mer fysikkforståelse ut av beregningsorientert fysikk. Det var spesielt etter andreårsemnet Introduksjon til vitenskapelige beregninger (vitber) at fysikken fikk komme i fokus. Vitber er egentlig et matematikkemne som handler om numerikk, men det inneholder også tre store programmeringsprosjekter som strekker seg over hele semesteret. M2 mener også det var vanskelig å se nytten av programmering i begynnelsen av studiet, det var først i andre semester da de begynte med numeriske prosjekter at han forstod hvor nyttig programmering er i fysikk. Årsaken er at det gir

mulighet til å plotte grafer av numeriske løsninger, som er vanskelig å gjøre med andre, enklere programmerer som Geogebra.

Kostnad

De tre intervjudeltakerne beskriver programmeringsoppgavene som mer krevende enn andre oppgaver. K2 sier at når de får en numerisk øving, må alt annet arbeid nedprioriteres og utsettes for at de numeriske øvingene skal bli gjort ferdig i tide. Og det er visst press på studentene for få å levert programmeringsøvingene i tide, siden de ellers mister muligheten til å gå opp til eksamen. Dette er ofte ikke tilfellet med de vanlige regneoppgavene, det til tross for at det kun er vanlige regneoppgaver som inngår i pensum til eksamen. Ifølge K2 er noen av faktorene som gjør det krevende at man ikke helt skjønner hva det blir bedt om, hva man skal finne ut av eller hvordan man skal programmere det, medlemmene i gruppa er uenige om hvordan svaret burde programmeres og at det er et stort antall oppgaver og deloppgaver. M2 mener at det tidvis har vært litt for stor arbeidsmengde. For hans egen del var det ikke så viktig siden han foretrekker programmeringsoppgaver, men det gikk ut over hans mulighet til å lære seg det eksamensrelaterte pensumet. K3 legger til at programmeringsoppgavene ofte er mer krevende siden man må sitte lenge og løse feilmeldinger.

Progresjon

I intervjuene forteller studentene ofte om hvordan de mener deres erfaringer med programmering i fysikk har utviklet seg i løpet av studiene, ofte med tanke på mestringsforventning, interesseverdi, nytteverdi eller kostnad. K2 beskriver en stor endring i løpet av studiet.

Meg: *[...] hvis du sammenligner programmering på en måte nå med på starten, hva er det som... som gjør at det er forskjellig?*

K2: *Ehm vet ikke. Det er vel... ehm... største forskjellen er vel at nå gruer meg mest til de, så på en måte har jeg har ikke noe lyst til å prøve på de heller, jeg har ikke lyst til å på en måte se på det en gang, jeg utsetter å se på de til jeg plutselig på en måte må se på det. Eehm mens i starten var det sånn «ok, la oss se på det, og gjøre noe med det også... sånn at det stemmer da.» Selv om man ikke har skjont fysikkdelen, så går det på en måte greit. Så hvis man ikke skjønner fysikkdelen, men skjønner*

programmeringsdelen, så går det på en måte opp i opp. Men nå er det mer sånn «Shit må jeg gjøre dette?» «Vil jeg gjøre dette?» Vet ikke fordelene... Eh og på en måte... det er ikke programmeringen i seg selv jeg ikke liker, men jeg har ikke lyst til å gjøre dette lenger, jeg gruer meg til jeg får numeriske oppgaver.

K2 uttrykker en sterkt redusert mestringsforventning og interesse som går ut over hennes motivasjon for å prøve å løse oppgavene. Samtidig legger hun til at hun anser programmering som svært nyttig der hun er nå, men at hun ikke anså det som særlig nyttig i begynnelsen av studiet.

M2 uttrykker også at han ser nytten av programmering nå i mye større grad enn i begynnelsen av studiet, da trodde han det kun dreide seg om å lage nettsider og spill. Ellers har ikke M2 sin holdning til programmering endret seg i løpet av studiet.

K3 sier også at hennes forståelse av nytteverdien i programmering har økt mye siden hun startet å studere. Da hun begynte å studere trodde hun programmering bare var noe man drev med innen rene IT-bransjer, men nå ser hun på det som svært nyttig i fysikk. I motsetning til K2, har K3 opplevd en sterk oppgang i mestring og interesse. Hun syntes det var tyngst i begynnelsen, siden det var en stor forskjell mellom programmeringen studentene lærte i programmeringsemnet ITGK og programmeringen studentene brukte i fysikkemnene. Siden har hun nådd et nivå innen programmering som gjør at programmeringen ikke oppleves som en utfordring i beregningsorientert fysikk.

Systemet rundt studentene

Så langt har jeg presentert de delene av intervjuene som handler om studentenes egen motivasjon og utvikling. Det er dog uunngåelig at samtaletemaet tidvis dreier seg over mot det som befinner seg rundt studentene, som for eksempel NTNU sitt utdanningssystem eller samarbeid og samspill med andre studenter. Et gjennomgående tema er samspillet mellom fysikkemnene og de rene programmeringsemnene. K2 og K3 er enige om at det er en forventning i fysikkemnene om at studentene skal ha lært nok programmering i programmeringsemnene til å kunne programmere i fysikk, selv om dette ikke alltid er tilfellet. De påstår at grunnkurset i programmering, ITGK, gir ferdigheter som er på et lavere nivå enn det som trengs i fysikken, og at deler av programmeringskompetansen som brukes i fysikk knapt undervises i ITGK i det hele tatt. K3 påpeker at det var først i fjerde semester i

emnet Introduksjon til vitenskapelige beregninger at hun begynte å oppleve at programmeringsoppgavene i fysikk var godt tilpasset opplæringen hun hadde fått, og det var da hun følte at «det løsnet».

På bakgrunn av dette savner K2 undervisning i fysikkemnene om programmeringsoppgavene, oppgavene baserer seg på i stor grad på at man finne ut av det selv. Studentene sier at de bruker mye tid på å søke opp programmeringstips på internett og synes at informasjonen de finner der ofte er nyttig. Andre kilder til hjelp eller informasjon er medstudenter, utdelt eksempelkode og studentassistenter. M2 og K3 understreker at det fungerer best å søke opp ting som dreier seg om programmering; dersom man mangler informasjon om fysikken eller matematikken, blir det vanskeligere å finne svarene på nettet. K3 mener at hun over tid synes det å finne hjelp på nettet er en grei ting, men at det i starten av studiet var frustrerende å måtte gjøre dette. Dersom studentene mangler for mye informasjon i begynnelsen av oppgaven, kan dette medføre at de har for mye de trenger å finne selv og dermed ikke makter å komme i gang med oppgaven.

K2 opplever at hun får lite eller ingen tilbakemelding på programmeringsoppgavene i etterkant, som gjør det vanskeligere å ta vite hva som gjort feil eller riktig og hva hun burde ta med seg videre.

Kjennetegn på en god programmeringsoppgave

Da studentene ble spurt om hva de mener kjennetegner en god programmeringsoppgave, svarte de at det er en oppgave som bidrar til å gjøre fysikken mer forståelig, det er en oppgave som er delt opp seksjoner eller bolker, den inneholder tips for å komme i gang med oppgaven, den inneholder informasjon om hvordan man skal plote svarene sine, den kan løses uavhengig av om man har løst foregående oppgave og den kan relateres til fysikkpensum ellers i emnet.

I løpet av intervjuene kommer studentene med syn på hva de tenker kunne vært gjort for å gjøre studieprogrammet sitt bedre med tanke på den beregningsorienterte fysikken. K2 og K3 nevner at det spesielt i første årstrinn kunne vært en ide å ha en innføring eller introduksjon i forkant av obligatoriske numeriske øvinger gjennom forelesninger eller valgfrie øvinger. M2 mener det viktigste som kan forbedres er å knytte programmeringsoppgavene mer opp mot det som er pensum ellers i faget, slik at man

opplever at timene som har gått til programmeringsøvinger også bidrar til å forberede en til eksamen.

M2 trekker fram andresemesteremnet Bølgefysikk og fluidmekanikk som et emne hvor programmeringsøvingene gav en relativt god læringsopplevelse, siden oppgavene der gav god opplæring i numeriske metoder. K3 peker ut tredjesemesteremnet Innføring i kvantefysikk siden det bidro til forståelse for fysikken ellers i emnet.

M2 synes samtidig at læringsopplevelsen av øvingene i Bølgefysikk og fluidmekanikk manglet relevans for fysikken ellers i emnet, som reduserte kvaliteten på læringsopplevelsen. K3 mener hun hadde en relativt dårlig læringsopplevelse med de beregningsorienterte fysikkoppgavene generelt i hele første årstrinn siden de ofte ikke lyktes med å løse oppgavene selv med hjelp studentassistenter.

6 Diskusjon og konklusjon

I dette kapitlet sees resultatene fra spørreskjemaundersøkelsen, intervjuene og relevant teori i lys av hverandre, for å sammenfatte en felles konklusjon som besvarer problemstillingen og forskningsspørsmålene. Diskusjonen tar utgangspunkt i de kvantitative resultatene og bruker kvalitative resultater til å utdype de kvantitative funnene og få dypere innsikt der det er aktuelt.

6.1 Studenters motivasjon i ulike faser av ferdighetsstrengen

Mestringsforventning

I EV-modellen defineres mestringsforventning som ens overordnede inntrykk av ens egen kompetanse innenfor et felt (Eccles og Wigfield, 2002). Innenfor beregningsorientert fysikk sammenfatter dette fysikk, matematikk og programmering. Indikatorer for mestringsforventning som jeg har brukt i mine undersøkelser retter seg mot i hvilken grad studenter forventer at de klarer å løse oppgaver, i hvilken grad de opplever å ha mestret tidligere oppgaver og i hvilken grad de synes tidligere oppgaver har vært vanskelige. I spørreskjemaet er det påstand 2.3 «Jeg forventer at jeg klarer å løse programmeringsoppgaver i fysikk.» som måler mestringsforventning, og den fikk gjennomsnittsscore 4,2, 3,6 og 3,9 for henholdsvis 1., 2. og 3. årstrinn (figur 5). Påstand 2.3 hadde markant høyere standardavvik i andre årstrinn enn i første og tredje årstrinn. Dette

tyder på at mange av studentene ankommer studiet med stor mestringsforventning, og i løpet av det første studieåret holder dette seg ganske likt for de fleste, men for noen studenter blir mestringsforventningen mye lavere. I tredje årstrinn ser det ikke til å være noen gruppe med stor reduksjon i mestringsforventning, men det er fremdeles i gjennomsnitt en litt lavere mestringsforventning i tredje årstrinn enn i første årstrinn. Det overordnede bildet er at beregningsorientert fysikk er vanskeligere enn studentene først hadde trodd, men at mange studenter lykkes med å innhente seg etter en tøff begynnelse. Dette bildet kan være et resultat av en skjevhet i utvalget, siden tredjeårsstudentene som hadde valgt matematikkretningen fikk kortere tid til å svare på spørreskjemaet, og har høyere sannsynlighet for å være underrepresentert. Eventuelt er det også mulig at studentene som får det tøft i første årstrinn bytter til andre studieprogrammer, og at det i hovedsak er de studentene som klarte seg i første årstrinn som er med videre i tredje årstrinn.

Resultatene fra den kvantitative analysen tyder på at mestringsforventningen er lavest i andre årstrinn ($s=3,6$) og at det der er størst forskjell mellom studentene får god støtte i de kvalitative analysene. Flere studenter peker på at grunnkurset i programmering (ITGK) som studentene har i første semester ikke gir opplæring i flere av de grunnleggende programmeringselementene i beregningsorientert fysikk. Dette viser også Haavind et al. (2021), der en stor andel av studentene svarer at de har lært seg å bruke biblioteker som er sentrale i beregningsorientert fysikk kun på egenhånd og at opplæringen har vært mangelfull. I frisvars spørsmålene nevner studentene et stort nivåsprang fra første til andre semester, spesielt emnet Elektrisitet og magnetisme i andre semester beskrives som å være utfordrende. Dette ser man igjen i vedlegg F1, F2 og F4, der Elektrisitet og magnetisme (elmag) konsekvent får en lavere score (høyere score der påstanden er negativ) enn de andre fysikkemnene. Student M2 fra intervjuene peker ut noen av programmeringsøvingene i elmag som de eneste programmeringsoppgavene han ikke har klart å løse. Også fysikkemnet i første semester, Mekanisk fysikk (mekfys), beskrives av flere som vanskelig siden nivået der er høyere enn i ITGK. Men K2 syntes programmeringsoppgavene i mekfys var overkommelige siden de var vant til å drive med programmering i ITGK og siden hun der kunne utnytte sitt forsprang i programmering fra videregående og dermed «kompensere» for at hun ikke alltid forstod fysikken. Relevansen av programmering i fysikk nevnes mye i de

kvalitative dataene, men det er vanskelig å se noen direkte kobling mellom det og mestringsforventning. Studenter kan lykkes godt med beregningsorientert fysikk uten å anse det som nevneverdig relevant for fysikkpensum ellers.

Mestringsforventning har signifikant korrelasjon i minst ett årstrinn med tre av de andre konstruktene: interesseverdi, kostnad og forhåndskunnskaper (tabell 8, 9, 10 og 11). For alle årstrinn er det signifikant korrelasjon mellom mestringsforventning og interesseverdi og kostnad, som innebærer at studenter som har høy mestringsforventning også synes programmering i fysikk er mer interessant og mindre krevende. Dette stemmer godt med resultatene fra Eliassen (2020). Det er logisk at studenter som opplever at de ikke får til oppgavene også opplever dem som mer krevende. Det er på annen side ikke like gitt at det skal være avgjørende at studentene må mestre oppgavene for at de skal like dem. Det vitner om at studentenes motivasjon i stor grad avhenger av hvordan de presterer og at deres interesse for programmering i fysikk er svært resultatdrevet. Denne teorien støttes av Bungum et al. (2012) sine resultater som viser at interesse og gode resultater er viktige motivasjonsfaktorer for fysikkstudenter.

Mestringsforventning og forhåndskunnskaper har signifikant korrelasjon i første årstrinn, men ikke i andre eller tredje årstrinn (tabell 9, 10 og 11). Dette skyldes antakeligvis at forskjellen i forhåndskunnskaper spiller størst rolle tidlig i studiet. Ut i andre og tredje årstrinn har forskjellene mellom studentenes nivå jevnet seg ut. Nytteverdi korrelerer lite med mestringsforventning, mer presist er det kun signifikant korrelasjon i tredje årstrinn. Dette kan henge sammen med at nytteverdi retter seg mot viktigheten av programmering i fysikk etter endte studier, og at tredjeårsstudentene er de som er nærmest å fullføre utdanningen blant studentene i utvalget.

Interesseverdi

Interesseverdi er definert som hvor mye interesse man tillegger noe. I spørreskjemaet består konstruktet interesseverdi av de to påstandene «Jeg er interessert i programmering generelt.» og «Jeg er interessert i programmering i fysikk.».

Fra figur 6 ser man at studentene generelt er interesserte i programmering (score ligger rundt 4) og at dette er ganske likt for alle årstrinn. Siden studentenes programmeringsferdigheter utvikler seg mye i løpet av disse årene, kan det virke

overraskende at interessen ikke endrer seg noe særlig i tillegg, spesielt hvis man sammenligner første og andre årstrinn, som har nærmest identiske gjennomsnittsverdier (blå søyler: $s=4,2$, grønne søyler: $s=3,9$).

Vedlegg F3 viser at studentene har en preferanse for vanlige fysikkoppgaver sammenlignet med programmeringsoppgaver i fysikk, som jevner seg ut i løpet av de to første studieårene. Vedlegg F4, som sammenligner studentenes mestring i programmeringsoppgaver i fysikk med vanlige fysikkoppgaver, har samme tendens som figur vedlegg F3. Dette styrker påstanden om at fysikkstudentenes interesse for programmering i fysikk er nært knyttet opp mot deres mestringsforventning. Dette kommer også frem i intervjuene, der studentene blant annet sier at det er gøy å lage grafer hvis grafen er riktig, og at det er gøy å lage animasjoner hvis animasjonene er korrekte. Studenten K2 påpeker dessuten at ITGK var lett og dermed også gøy, mens når programmeringsoppgavene er vanskeligere, har hun ikke lyst til å se på dem engang. K3 mener at mestringsfølelsen man får av å løse oppgaver er en av de viktigste drivkreftene for fysikkstudenter generelt. Det innebærer også at hvis studentene mister troen på at de kan løse oppgavene, kan det få store konsekvenser for deres generelle motivasjon på studieprogrammet.

Interesseverdi og nytteverdi korrelerer ikke med hverandre i første eller andre årstrinn, mens i tredje årstrinn korrelerer noen av konstruktene og påstandene med hverandre (tabell 9, 10 og 11). Dette tyder på at det tidlig i fysikkstudiene ikke er noen sammenheng mellom studentenes syn på programmeringens fremtidige nytte i arbeidslivet og deres egen interesse for programmering. En mulig forklaring er at studentene har lite informasjon om hvordan programmering brukes i arbeidslivet og ikke vet om de vil finne den interessant på samme måte som den programmeringen de har i studiene. Resultatene fra frisvarspørsmålene viser at det er mulig at studentene anser programmering som nyttig i framtiden, selv om de ikke vet hvordan eller hvorfor.

Interesseverdi og kostnad korrelerer på tvers av alle årstrinn; studenters som synes programmering i fysikk er interessant, synes også at det er mindre krevende. Det kan forstås i lys av øvingsstrukturen i mange av fysikkemnene, der programmeringsoppgavene er obligatoriske å bestå for å kunne gå opp til eksamen. For studenter som ikke er så interessert i programmering, blir programmeringsøvingene en ekstra byrde hvor målet er å legge inn minst mulig innsats for å ha mest mulig tid til de analytiske, eksamensrelaterte

fysikkoppgavene. Ikke alle fysikkstudenter anerkjenner beregningsorientert fysikk som en del av fysikken, men mener heller at den analytiske fysikken er den ekte fysikken. Studenten M2 foretrekker derimot beregningsorientert fysikk fremfor analytisk fysikk og har derfor ikke noe problem med at programmeringsøvingene går ut over eksamensforberedelsene.

Interesseverdi og forhåndskunnskaper korrelerer i første årstrinn i alle kombinasjoner av konstrukter og påstander, mens i andre og tredje årstrinn er det ingen korrelasjon, bortsett fra i en kombinasjon i tredje årstrinn. Som med mestringsforventning er det logisk at studentenes forhåndskunnskaper har størst påvirkning på interesseverdien til studentene i første årstrinn. I andre og tredje årstrinn er ikke lenger interessen for programmering avhengig av hva studentene kunne om programmering i forkant av studiene, som vitner om at studentenes forhåndskunnskaper spiller en mindre rolle lenger ut i studiene. Eksempler fra intervjuene peker også i denne retningen, siden studenten K2 som allerede kunne programmering ved studiestart, ikke lenger likte det, mens K3 hadde ingen programmeringskunnskaper ved studiestart, men var likevel svært positiv til programmering i intervjuet.

Interesseverdi og IMPEL-påstanden «Jeg lærer fysikk godt av å programmere.» har høy korrelasjon i første og tredje årstrinn (henholdsvis $p=0,561$ og $p=0,460$), som tyder på at studentene i hovedsak ser på programmering som et redskap for å lære seg fysikk. Studenters interesse for beregningsorientering kan økes hvis studentene opplever at de kan bruke programmering i emner de interesserer seg for (Eliassen, 2020). I andre årstrinn derimot er det ikke signifikant korrelasjon, som tyder på deres interesse for programmering er uavhengig av hvorvidt de lærer fysikk av å drive med programmering. En mulig forklaring på at det ikke er korrelasjon i andre årstrinn er at interessen for programmering er høyere enn interessen for fysikk, slik at det ikke er så viktig for studentene om de lærer fysikk av å programmere. Figur 6 viser at interesseverdien for programmering er ganske jevn for alle årstrinn, så det vil i så fall innebære at andreårsstudentene har lavere interesseverdi for fysikk enn øvrige årstrinn. I tredje årstrinn kan det ha vært en skjevhet i utvalget i favør studenter som er interessert i fysikk (det er for eksempel grunn til å tro at en matematikkstudent vil være mindre motivert for å svare på et spørreskjema som heter «Programmering i fysikk» enn en fysikkstudent, dessuten fikk matematikkstudentene

kortere tid til å svare (delkapittel 4.1.3)). Denne skjevheten ville medføre en økt korrelasjon mellom interesseverdi for programmering og å lære fysikk godt av å programmere.

Nytteverdi

Konstruktet nytteverdi dreier seg om hvorvidt studentene ser på programmering som noe de får bruk for. Resultatene fra dette konstruktet skiller seg ut fra de andre konstruktene av flere årsaker. Figur 7 viser at påstandene i nytteverdikonstruktet «Programmering er viktig hvis jeg skal bli forsker.» og «Programmering er viktig i arbeidslivet.» får svært høy score i alle årstrinn, og det er knapt noen forskjell mellom årstrinnene. I spørreskjemaet skiller dette konstruktet seg fra andre konstrukt ved at det kun setter søkelys på tiden etter studiene. I intervjuene trakk alle studentene frem at programmering har stor nytteverdi med tanke på arbeidslivet, som stemmer godt med resultatene fra spørreundersøkelsen.

Studentene i intervjuene pekte på andre sider av begrepet nytte; sider som er relatert til fysikkstudiene. Dette er en annen måte å forstå nytte på som fanges mer opp av IMPEL-påstanden «Jeg lærer fysikk godt av å programmere.», nemlig at programmering er nyttig som et verktøy i fysikk. Der er svarene langt mer varierte og ligger nærmere midten av svarskaalen («Verken enig eller uenig»). M2 sier at han i første semester ikke forstod hvorfor man skulle bruke programmering i fysikk fremfor matematiske dataprogrammer. Det var først i andre semester at han innså den nytten av å ha et eget program som gjør de beregningene man ønsker. Nordby (2019) beskriver også hvordan man kan ha redusert forståelse av nytte siden det i begynnelsen ikke er åpenbart hvordan programmering er egnet for fysikk.

Svarene på IMPEL-påstanden fra IMPEL-undersøkelsen høst 2019 kan sammenlignes med svarene på samme påstande i denne undersøkelsen. Studentene som svarte i IMPEL-prosjektet høst 2019, gikk i første årstrinn høsten 2019. Det er de samme studentene som går i tredje årstrinn i min undersøkelse, med unntak av studentene fra lektorutdanning i realfag. Blant denne studentgruppen har det opplevde fysiske læringsutbyttet i programmering økt fra å ligge på 2,6 til å ligge på 3,6, altså omtrent et nivå opp på Likertskaalen. Det er logisk at studentene i prosessen med å lære seg mer programmering og mer beregningsorientert fysikk, også opplever økt læringsbytte av fysikk i programmering. Til gjengjeld må man igjen ta i betraktning faren for en skjevhet i utvalget blant tredjeårsstudentene i min undersøkelse. I min spørreskjemaundersøkelse er også utvalget

(N=37) mindre enn i IMPEL-undersøkelsen (N=158). Hvis man sammenligner førsteårsstudentene i 2019 med førsteårsstudentene i 2021, gir 2019-resultatene et noe lavere snitt ($s=2,9$ mot $s=2,6$). Dette passer med dataene som viser at 2021 første årstrinn hadde større forhåndskunnskaper enn 2021 tredje årstrinn og at det er positiv korrelasjon mellom forhåndskunnskaper og IMPEL-påstanden i første årstrinn ($p=0,387$). Denne konklusjonen har dog relativt stor usikkerhet, siden IMPEL-spørreundersøkelsen har en annen struktur og utforming enn spørreundersøkelsen i denne oppgaven og ulik størrelse på utvalget (N=158 i IMPEL og N=57 i dette masterprosjektet). Den lavere scoren fra IMPEL-undersøkelsen kan skyldes at utvalget der også inkluderte studentene på lektorutdanning i realfag, men ikke i min undersøkelse.

Konstruktet nytteverdi korrelerer svært lite med andre konstrukter, spesielt i første og andre årstrinn. Blant motivasjonsfaktorer testet i Bungum et al. (2012), har yrkesmuligheter relativt lav score, som tyder på at det er lite sammenheng mellom fysikkstudenters motivasjon og syn på nytten de vil ha i arbeidslivet av det de lærer.

I første årstrinn er det ingen signifikant korrelasjon mellom nytteverdi og andre konstrukter, i andre årstrinn korrelerer nytteverdi kun med påstanden om at man lærer fysikk godt av å programmere ($p=0,341$). Studenter i andre årstrinn som anser programmering som nyttig i fremtiden, mener altså også at de lærer fysikk godt av å programmere. Det kan tyde på at de i større grad ser for seg hvordan de kan bruke beregningsorientert fysikk i arbeidslivet.

Det er korrelasjon mellom konstruktet forhåndskunnskaper og påstand 2.7 som inngår i nytteverdikonstruktet ($p=0,383$). Dette tyder på at studenter som allerede har lært seg programmering i forkant av studiene, også i større grad har en oppfatning av at programmering er viktig i arbeidslivet. De tre intervjudeltakernes bakgrunn peker også i den retning. Det er vanskelig å avgjøre om det var holdningen om at programmering er viktig i arbeidslivet som gjorde at studentene valgte å lære seg programmering før studiene eller om programmeringsopplæringen før studiene førte til en økt holdning av at programmering er viktig i arbeidslivet, eller om det var begge deler.

I tredje årstrinn er det korrelasjon mellom nytteverdi og mestringsforventning. Altså har studenter som mestrer programmering i fysikk en større oppfatning om at dette er noe de vil bruke i fremtidig jobb. En mulig forklaring på dette er at studentene i tredje årstrinn har

begynt å prøve relevante sommerjobber, som K3 fra intervjuene er et eksempel på. En annen forklaring er at studentene er nærmere endte studier og derfor har begynt å finne mer ut av hva de vil jobbe med.

Kostnad

Kostnadskonstruktet er et negativt konstrukt som måler hvor krevende studentene synes beregningsorientert fysikk er. I denne oppgaven er det påstanden «Programmeringsoppgaver i fysikk er slitsomme/krevende sammenlignet med andre oppgaver.» som representerer konstruktet kostnad. Figur 8 viser at påstanden scorer mellom 3 og 4, altså er de fleste studenter enten litt enig eller verken enig eller uenig. I førsteåret er studentene mest nøytrale ($s=3,0$), i andreåret er studentene minst nøytrale ($s=3,5$). Med et standardavvik på omtrent 1,2 virker det å være stor forskjell mellom studentene. Siden påstanden sammenligner programmeringsoppgaver med vanlige oppgaver, betyr disse resultatene i hovedsak av flertallet av studentene synes programmeringsoppgavene er like krevende eller litt mer krevende enn vanlige oppgaver. Resultatene forteller ikke i hvilken grad studentene synes begge typer oppgaver er krevende, som er en svakhet ved å ha et sammenlignende spørsmål. M2, K2 og K3 fra intervjuene er imidlertid enige om at programmeringsoppgaver generelt krever større arbeidsmengde enn vanlige oppgaver og at de tar mye tid. K2 hevder at hun er nødt til å sette alt annet til side når de får en programmeringsøving for å klare å gjøre den ferdig i tide. I frisvarspørsmålene nevner flere studenter, derav spesielt andreårsstudenter, at nivået i programmeringsoppgavene er for høyt og at de krever ferdigheter som de ikke har lært. Dette sammen med den reduserte mestringsforventningen i andre årstrinn støtter funnet fra prosjektoppgaven forrige vår om at det er kunnskapshull i beregningsorientert fysikk.

De fleste korrelasjoner som inkluderer konstruktet kostnad er allerede nevnt. Kostnad korrelerer med mestringsforventning og interesseverdi, men ikke med nytteverdi. I første og tredje årstrinn korrelerer kostnad med forhåndskunnskaper, men ikke i andre årstrinn. Det er logisk at studenter med forhåndskunnskaper i programmering opplever det som mindre krevende i første årstrinn, men i tredje årstrinn skulle man tro at det ikke spiller noen rolle. Det virker altså her som om forkunnskapene kan ha påvirkning langt inn i studiene, til tross for andre resultater, som at mestringsforventning bare korrelerer med forhåndskunnskaper i

første årstrinn ($p=0,505$). Det er uklart hvordan man skal tolke disse motstridende resultatene.

6.2 Anbefalinger for videreutvikling av ferdighetsstrengen

Resultatene fra denne masteroppgaven gir grunnlag for å vurdere hva ferdighetsstrengen lykkes godt med og hvordan den kan utvikles videre. En av hovedideene bak ferdighetsstrengen er å knytte fysikkemnene mer sammen og å gjøre overgangen dem imellom jevnere. Store hopp fra ett emne til det neste øker sannsynligheten for at studentene opplever redusert mestringsforventning. Som resultatene viser spiller mestringsforventningen en stor rolle for studentenes helhetlige inntrykk og motivasjon for oppgavene. Ferdighetsstrengen inkluderer kun programmeringen som inngår i fysikkemnene, men studentenes erfaringer med programmering er også sterkt påvirket av andre emner som inneholder programmering. Det mest nevnte eksemplet er ITGK; introduksjonskurset i programmering som studentene har i første semester. Flere studenter mener at dette faget ikke gir den opplæringen som trengs i programmering for å drive med beregningsorientert fysikk. Derfor har studentene utilstrekkelig programmeringskompetanse helt fra begynnelsen av og flere legger inn en stor innsats for å nå det forventede nivået i beregningsorienteringen i fysikkundervisningen. Derfor er den viktigste anbefalingen fra denne studien at studentene må få en bedre tilpasset programmeringsopplæring helt i begynnelsen av studiet. Økt fokus på beregningsorientering har gode pedagogiske og faglige argumenter (Malthe-Sørensen et al., 2015). Enten burde ITGK endres eller eventuelt erstattes av et eget introduksjonskurs i programmering for fysikkstudenter, eller så burde det være økt opplæring i beregningsorientert fysikk i fysikkemnene.

Studentenes syn på nytten av programmering i fremtidig arbeidsliv og forskning er at den er svært høy, samtidig har den svært lite påvirkning på studentenes mestring og interesse. På bakgrunn av dette anbefales det å knytte beregningsorientert fysikk i fysikkstudieprogrammer mer opp mot beregningsorientert fysikk slik den brukes i forskning og arbeidsliv, slik at studentene forstår bedre hvorfor de lærer det de gjør.

Beregningsorientert fysikk minner mer om fysikken man finner i arbeidsliv og forskning, dermed vil økt fokus på beregningsorientering i fysikkutdanningen også bidra til å få fysikken til å virke mer relevant for studentene. Programmering er i seg selv en stadig mer ettersøkt ferdighet i arbeidsmarkedet (Dysthe et al., 2021; OECD, 2019).

Det framkommer spesielt i intervjuene og frisvarspørsmålene at fysikkemnenes struktur reduserer studentenes motivasjon for å drive med beregningsorientert fysikk. Dette skyldes ifølge studentene at programmeringsoppgavene ikke virker relevant for fysikken ellers i fysikkemnene. Hvis beregningsorientert fysikk hadde vært mer integrert i fysikkpensum, ville studentene også hatt større nytte av å legge inn best mulig innsats i programmeringsoppgavene. Dette kan oppnås ved å for eksempel inkludere beregningsorientering i forelesninger og andre læringsaktiviteter. Det at beregningsorientering ikke oppleves som integrert i fysikkemnene kan også knyttes til at beregningsorientert fysikk er obligatorisk for å få gå opp til eksamen, men det er ikke en del av det pensumet som inngår på eksamen. Hvis beregningsorientering også inngår i eksamenspensum og dermed påvirker karakteren, kan det øke virke mer relevant for emnet og dermed øke studentenes motivasjon. Slik det er nå, er det en fare for at beregningsorientering i fysikk havner på siden av det studentene opplever som det faget egentlig skal handle om og mister derfor verdi i studentenes øyne.

6.3 Forslag for videre forskning

Denne undersøkelsen har gitt resultater angående sammenhenger mellom studenters mestringsforventning, interesseverdi, nytteverdi og kostnad innen beregningsorientering i fysikkutdanningen. Det er dog en viss usikkerhet i resultatene knyttet til undersøkelsenes validitet og den begrensede tidsperioden som utgjør rammebetingelsen for en masteroppgave. For videre forskning anbefales det derfor å søke dypere innsikt i studenters motivasjon i ulike stadier av fysikkstudiet ved å følge en og samme studentgruppe over tid. Dette vil bidra til å svare på en rekke spørsmål som kan stilles i kjølvannet av denne masteroppgaven. Hvordan påvirker det fysikkstudenter når de får redusert mestringsforventning? For å svare på dette spørsmålet ville det være passende å gjennomføre en kvalitativ studie som følger et lite utvalg studenter fra studiestart gjennom det første studieåret. Hva er sammenhengen mellom mestringsforventning og frafall? Her ville det være aktuelt å gjøre en kvantitativ undersøkelse som måler mestringsforventning til studenter som faller fra og sammenligner med den resten av studentpopulasjonen. Hva er årsaken til at tredjeårsstudenter har høyere mestringsforventning enn andreårsstudenter i beregningsorientert fysikk? En kvalitativ studie som følger studenter gjennom det andre studieåret, vil bidra til å svare på dette spørsmålet. Hvordan kan man lage

beregningsorienterte fysikkoppgaver som er utfordrende uten at det går ut over studenters motivasjon? Her ville det være aktuelt å utforme alternative øvingsopplegg og undersøke studenters erfaringer knyttet til mestring, interesse, nytte og kostnad før og etter at de har prøvd å løse øvingene.

6.4 Konklusjon

Problemstillingen og forskningsspørsmålene i denne oppgaven dreier seg om studenters erfaringer med beregningsorientert fysikk i de tre første studieårene. Mer konkret er studentenes motivasjon delt inn i de fire underkategoriene mestringsforventning, interesseverdi, nytteverdi og kostnad. Etter å ha tatt for seg hver dimensjon i EV-modellen og undersøkt disse i ulike årstrinn, er det noen tydelige funn som peker seg ut. Alle konstrukter scorer jevnt over ganske godt (fra 3 og oppover) i samtlige årstrinn, det er ingen påstander direkte knyttet til motivasjon som fikk veldig lav score (under 2). Studentene anser programmering i fysikk som svært nyttig for fremtiden i alle årstrinn. Studentene i alle årstrinn viser også høy interesse for beregningsorientert fysikk. Mestringsforventning scorer høyere enn 3 i alle årstrinn, men scorer lavere i andre årstrinn enn i første og tredje årstrinn. Studentene i andre årstrinn synes også at beregningsorientering er mer krevende enn øvrige studenter.

Noen forklaringer fra de kvalitative dataene på den reduserte mestringsforventningen som er funnet i andre årstrinn, er at opplæringen studentene får i ITGK og i fysikkemner ikke forbereder dem godt nok til å løse programmeringsoppgavene i fysikk. Det gjør at oppgavene oppleves som mer krevende og reduserer studentenes mestringsforventning. Siden mestringsforventning, interesseverdi og kostnad er konstrukter som korrelerer mye med hverandre, er det grunn til å tro den manglende opplæringen går ut over flere studenters totale motivasjon. Flere studenter peker ut fysikkemnene studentene har i andre semester som spesielt krevende. Et sentralt spørsmål som resultatene fra denne oppgaven kun gir indikasjoner på, er hvorfor tredjeårsstudenter gir inntrykk av på være mer motivert for beregningsorientert fysikk enn andreårsstudenter. Mulige teorier er at det kan være en skjevhet i utvalget som er spesielt stor i tredje årstrinn, eller at studentene i løpet av andre årstrinn får et bedre forhold til programmering blant annet på grunn av fjerdesemesteremnet Introduksjon til vitenskapelige beregninger, eller at studenter slutter på fysikkstudiet som konsekvens av redusert motivasjon.

7 Referanseliste

- Bennedsen, J. & Caspersen, M. (2007). Failure rates in introductory programming. *SIGCSE Bulletin*, 39, 32-36. <https://doi.org/10.1145/1272848.1272879>
- Braun, V. & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2), 77-101. <https://doi.org/10.1191/1478088706qp063oa>
- Brown, J. S., Collins, A. & Duguid, P. (1989). Situated Cognition and the Culture of Learning. *Educational Researcher*, 18(1), 32-42. <https://doi.org/10.3102/0013189X018001032>
- Bungum, B., Hauge, H. & Rødseth, S. (2012). Fysikkstudenten fra studiestart til mastergrad
Motivasjon, verdier og prioriteringer. *Uniped*, 35(3), 3-15.
<https://doi.org/10.3402/uniped.v35i3.19886> ER
- Dolonen, J. A., Kluge, A., Litherland, K. & Mørch, A. I. (2019). *Litteraturgjennomgang av programmering i skolen*. Universitetet i Oslo.
- Dysthe, D. K., Johnsen, S. G., Hjorth-Jensen, M., Lervåg, K. Y., Schjølberg-Henriksen, K., Simonsen, I. & Skjetne, P. (2021). *Evaluering av NTNUs studieprogram MTFYMA og BFY*. NTNU.
https://i.ntnu.no/c/wiki/get_page_attachment?p_l_id=1307200529&nodeId=1306956301&title=Utdanningskvalitet+-+NV&fileName=Rapport%20BFY%20MTFYMA%202020.pdf
- Eccles, J. & Wigfield, A. (2002). Motivational Beliefs, Values and Goals. *Annual Review of Psychology*, 53, 109-132. <https://doi.org/10.1146/annurev.psych.53.100901.135153>
- Eliassen, J. E. (2020). *Biologistudenters motisasjon for beregningsorientert biologi etter innføring av krav om full fordypning i realfaglig matematikk* [Universitetet i Oslo].
- Field, A. (2009). *Discovering statistics using SPSS* (3rd ed. utg.). SAGE.
- Gliem, J. A. & Gliem, R. R. (2003). *Calculating, Interpreting, and Reporting Cronbach's Alpha Reliability Coefficient for Likert-Type Scales*
- Haavind, H. H., Hansli, Å. C., Kristensen, A. D., Leraand, K. & Undheim, V. (2021). *Evaluering av lab og numerikk*. NTNU.
https://i.ntnu.no/c/wiki/get_page_attachment?p_l_id=1307200529&nodeId=1306956301&title=Utdanningskvalitet+-+NV&fileName=Rapport%20studentkomite%20BFY%20MTFYMA%202020.pdf

- Malthe-Sørenssen, A., Hjorth-Jensen, M., Langtangen, H. P. & Mørken, K. (2015). Integrasjon av beregninger i fysikkundervisningen. *Uniped*, 38(4), 303-310.
<https://doi.org/10.18261/ISSN1893-8981-2015-04-06> E
- Nordby, S. T. (2019). *Programmering og algoritmisk tenkning i fysikkundervisning* [Master, NTNU].
- NTNU. *Retningslinje for behandling av personopplysninger*. <https://innsida.ntnu.no/wiki/-/wiki/Norsk/Retningslinje+for+behandling+av+personopplysninger>
- NTNU. (2021, 13.05). *Fysikk og matematikk - Studiets oppbygning*. NTNU.
<https://www.ntnu.no/studier/mtfyma/oppbygning#year=2019&programmeCode=MTFYMA>
- Odden, T. O., Lockwood, E. & Caballero, M. (2019). Physics computational literacy: An exploratory case study using computational essays. *Physical Review Physics Education Research*, 15(2), 020152. <https://doi.org/10.1103/PhysRevPhysEducRes.15.020152>
- OECD. (2019). *PISA 2018 Results (Volume I): What Students Know and Can Do*.
<https://www.oecd.org/education/pisa-2018-results-volume-i-5f07c754-en.htm>
- Postholm, M. B. & Jacobsen, D. I. (2018). *Forskningsmetode for masterstudenter i lærerutdanning* (1. utgave. utg.). CAPPELEN DAMM AS.
- Robson, C. & McCartan, K. (2017). *Real World Research, 4th Edition*. Wiley.
- Shute, V. J., Sun, C. & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, 22, 142-158.
- Stormo, A. (2009). *Integrering av numeriske beregninger i grunnleggende fysikk-kurs* [NTNU]. Trondheim.
- Sørby, S. & Angell, C. (2012). *Undergraduate students' challenges with computational modelling in physics*.
- Utdanningsdirektoratet. (2021). *Kompetansemål etter fysikk 1*.
<https://www.udir.no/lk20/fys01-02/kompetansemaal-og-vurdering/kv466>
- Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L. & Wilensky, U. (2016). Defining Computational Thinking for Mathematics and Science Classrooms. *Journal of Science Education and Technology*, s. 127-147.
- Wigfield, A., Tonks, S. & Klauda, S. L. (2009). Expectancy-value theory. I K. R. W. A. Wigfield (Red.), *Educational psychology handbook series. Handbook of motivation at school* (s. 55-75). Routledge.

Wilson, B. C. & Shrock, S. (2001). Contributing to success in an introductory computer science course: a study of twelve factors. *SIGCSE Bull.*, 33(1), 184–188.

<https://doi.org/10.1145/366413.364581>

8 Vedlegg

Vedlegg A: Spørreskjema

Programmering i fysikk

Side 1

Obligatoriske felter er merket med stjerne *

Velkommen til denne spørreundersøkelsen som inngår i mitt masteroppgaveprosjekt!

Hvis ingen svaralternativer passer, er det best om du går videre uten å svare.

Vennlig hilsen Henrik Storesund, Institutt for fysikk.

Linje *

MTFYMA

BFY

Kjønn *

Kvinne

Mann

Annet

Trinn *

1. klasse

2. klasse

3. klasse

Hvordan fungerer programmeringsøvinger i fysikkemner for deg? *

Obligatoriske felter er merket med stjerne *

Hvor enig er du i følgende påstander?

	Helt uenig	Litt uenig	Verken enig eller uenig	Litt enig	Helt enig
Jeg er interessert i programmering generelt.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Jeg er interessert i programmering i fysikkemner.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Jeg forventer at jeg klarer å løse programmeringsoppgaver i fysikk.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Hvorvidt jeg liker en programmeringsoppgave, avhenger av i hvilken grad jeg klarer å løse den.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Jeg lærer fysikk godt av å programmere.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Programmering er viktig hvis jeg skal bli forsker.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Programmering er viktig i arbeidslivet.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Før jeg begynte på dette studiet, kunne jeg bruke if/else-setninger.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Før jeg begynte på dette studiet, kunne jeg bruke for/while-løkker.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Før jeg begynte på dette studiet, kunne jeg bruke numpy-biblioteket i Python.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Programmeringsoppgaver i fysikk er slitsomme/krevende sammenlignet med andre fysikkoppgaver.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Det er verdt innsatsen å arbeide med programmeringsoppgaver i fysikk.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Obligatoriske felter er merket med stjerne *

i Dette elementet vises kun dersom alternativet «2. klasse» eller «3. klasse» er valgt i spørsmålet «Trinn»

Videre kommer en rekke påstander hvor du skal huke av for grad av enighet for hvert enkelt fysikkemne.

i Dette elementet vises kun dersom alternativet «2. klasse» er valgt i spørsmålet «Trinn»

Jeg hadde fått god nok opplæring i programmering til å løse programmeringsøvingene.

	Helt uenig	Litt uenig	Verken enig eller uenig	Litt enig	Helt enig
FY1001 Mekanisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
FY1003 Elektrisitet og magnetisme	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4163 Bølgefysikk og fluidmekanikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

i Dette elementet vises kun dersom alternativet «2. klasse» er valgt i spørsmålet «Trinn»

Programmeringsøvingene krevde numerikkferdigheter som jeg ikke hadde lært.

	Helt uenig	Litt uenig	Verken enig eller uenig	Litt enig	Helt enig
FY1001 Mekanisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
FY1003 Elektrisitet og magnetisme	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4163 Bølgefysikk og fluidmekanikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

i Dette elementet vises kun dersom alternativet «2. klasse» er valgt i spørsmålet «Trinn»

Jeg mestret programmeringsøvingene bedre enn regneøvingene.

	Helt uenig	Litt uenig	Verken enig eller uenig	Litt enig	Helt enig
FY1001 Mekanisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
FY1003 Elektrisitet og magnetisme	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4163 Bølgefysikk og fluidmekanikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

i Dette elementet vises kun dersom alternativet «2. klasse» er valgt i spørsmålet «Trinn»

Jeg foretrakk regneøvinger fremfor programmeringsøvinger.


	Helt uenig	Litt uenig	Verken enig eller uenig	Litt enig	Helt enig
FY1001 Mekanisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
FY1003 Elektrisitet og magnetisme	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4163 Bølgefysikk og fluidmekanikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

i Dette elementet vises kun dersom alternativet «3. klasse» er valgt i spørsmålet «Trinn»

Jeg hadde fått god nok opplæring i programmering til å løse programmeringsøvingene.


	Helt uenig	Litt uenig	Verken enig eller uenig	Litt enig	Helt enig
FY1001 Mekanisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
FY1003 Elektrisitet og magnetisme	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

TFY4163 Bølgefysikk og fluidmekanikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4165 Termisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4215 Innføring i kvantefysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4195 Optikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

 Dette elementet vises kun dersom alternativet «3. klasse» er valgt i spørsmålet «Trinn»

Programmeringsøvingene krevde numerikkferdigheter som jeg ikke hadde lært.

	Helt uenig	Litt uenig	Verken enig eller uenig	Litt enig	Helt enig
FY1001 Mekanisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
FY1003 Elektrisitet og magnetisme	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4163 Bølgefysikk og fluidmekanikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4165 Termisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4215 Innføring i kvantefysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4195 Optikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

 Dette elementet vises kun dersom alternativet «3. klasse» er valgt i spørsmålet «Trinn»

Jeg mestret programmeringsøvingene bedre enn regneøvingene.

	Helt uenig	Litt uenig	Verken enig eller uenig	Litt enig	Helt enig
--	------------	------------	-------------------------	-----------	-----------

FY1001 Mekanisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
FY1003 Elektrisitet og magnetisme	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4163 Bølgefysikk og fluidmekanikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4165 Termisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4215 Innføring i kvantefysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4195 Optikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



Dette elementet vises kun dersom alternativet «3. klasse» er valgt i spørsmålet «Trinn»

Jeg foretrakk regneøvinger fremfor programmeringsøvinger.

	Helt uenig	Litt uenig	Verken enig eller uenig	Litt enig	Helt enig
FY1001 Mekanisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
FY1003 Elektrisitet og magnetisme	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4163 Bølgefysikk og fluidmekanikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4165 Termisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4215 Innføring i kvantefysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4195 Optikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Obligatoriske felter er merket med stjerne *



Dette elementet vises kun dersom alternativet «2. klasse» er valgt i spørsmålet «Trinn»


Hva er din gjeldende karakter i følgende emner:

	A, B eller C	D eller E	F
TDT4110 Informasjonsteknologi, grunnkurs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
FY1001 Mekanisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
FY1003 Elektrisitet og magnetisme	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4163 Bølgefysikk og fluidmekanikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Hva er din gjeldende karakter i følgende emner:

	A, B eller C	E eller D	F
TDT4110 Informasjonsteknologi, grunnkurs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
FY1001 Mekanisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
FY1003 Elektrisitet og magnetisme	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4163 Bølgefysikk og fluidmekanikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4165 Termisk fysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4215 Innføring i kvantefysikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TFY4195 Optikk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Har du noen kommentarer til denne spørreundersøkelsen eller andre kommentarer angående programmering i fysikk?

 Sideskift

Side 5

Obligatoriske felter er merket med stjerne *

Takk for dine svar!

Hvis du kunne tenke deg å delta på et kort intervju (ca 30 minutter) for å gi mer utfyllende svar om noen av disse temaene i november, kan du sende mail til min mailadresse: hensto@stud.ntnu.no.

Vedlegg B: Intervjuguide

Intervjuguide studenter ved fysikk og matematikk

Jeg vil ønske velkommen til dette intervjuet!

(Hensikten med dette intervjuet er å undersøke ditt forhold til programmering, primært innen fysikk. Det gjelder hva som gir deg motivasjon for å programmere, hva du synes om undervisningen her på NTNU og hvordan ditt syn på programmering har utviklet seg underveis på studiet.)

Som du helt sikkert er oppmerksomme på, er du her frivillig. Det vil si at du også bestemmer hva du ønsker å prate om, og når du ønsker å avslutte. Du kan i ettertid kreve å trekke deg som forskningsdeltakere, uten å oppgi noen grunn. De innsamlede dataene vil bli anonymisert.

Resultatene fra dette intervjuet vil inngå som informasjonsgrunnlag i min masteroppgave om beregningsorientert fysikk på universitetsnivå. Publiserte resultater fra dette intervjuet og andre intervjuer vil anonymiseres, slik at det ikke er mulig for andre enn meg å spore dine uttalelser tilbake til deg. Sporingen vil være avhengig av en koblingsnøkkel, en fil som lagres på én separat enhet. Etter prosjektets slutt vil denne koblingsnøkkelen bli slettet, slik at intervjuet på ingen måte kan spores tilbake til dere.

Jeg ønsker å bruke lydopptaker og håper at det er greit for deg. Intervjuet er ment å vare i om lag 30 minutter, men dette er avhengig av hvor mye du har på hjertet.

Mitt mål er å få deg til å dele så mye som mulig innenfor relevante temaer. Det er ikke meningen at det er jeg som sitter og prater mesteparten av tiden. For å bidra til dette, vil jeg sitte og lytte og skyte inn spørsmål der jeg mener det er passende. Det kan være for å gå dypere inn på temaer du har trukket fram eller bytte til andre temaer jeg ønsker å ta opp. Dine synspunkter vil være verdifulle med tanke på å lære mer om hvordan det numeriske opplegget i fysikk fungerer og å kunne tilrettelegge numeriske øvinger for fremtidige studenter.

- Vi kan starte med at du forteller kort hvem du er og hvilken erfaring du hadde med programmering før du begynte å studere ved NTNU.
- Hvordan er ditt forhold til programmering nå sammenlignet med da du begynte å studere?
 - o Hva synes du om programmeringsoppgavene du har nå?
 - o Hva syntes du om programmeringsoppgavene du hadde på starten?
- Hva kjennetegner en god programmeringsoppgave i fysikk?
- Hva er det som motiverer deg til å bruke programmering i fysikkemner?
- Hva tenker du om nytten av programmering for deg selv?
 - o Har dette inntrykket endret seg i løpet av studiene?
- Hva er ditt helhetsinntrykk av programmeringsoppgavene i fysikk sammenlignet med vanlige regneoppgaver?
 - o Vanskelighetsgrad?
 - o Arbeidsmengde?
 - o Hvor godt tilpasset de er?
 - o Har dette inntrykket endret seg i løpet av studiet?
 - o I hvilken grad synes du at kravene i programmeringsoppgavene er rimelig?
 - o Hvordan påvirker det deg når du er nødt til å lære deg ting på egenhånd i programmeringsoppgaver?
- Er det noen fysikkemner hvor du mener programmeringsoppgavene har gitt deg en spesielt god læringsopplevelse?
 - o Hvorfor/hvordan?
- Er det noen fysikkemner hvor du mener programmeringsoppgavene har gitt deg en spesielt dårlig læringsopplevelse?
 - o Hvorfor/hvordan?
- Er det noe du mener burde nevnes, men som ikke har blitt tatt opp i dette intervjuet?

Vedlegg C: Samtykkeerklæring

Vil du delta i masteroppgaven "Beregningsorientering i fysikkutdanningen"?

Dette er et spørsmål til deg om å delta i et masterprosjekt hvor formålet er å undersøke hvordan programmeringsoppgaver passer i fysikkemner på fysikk og matematikk-studiet. I dette skrivet gir vi deg informasjon om målene til oppgaven og hva deltakelse vil innebære for deg.

Formål

Formålet med dette forskningsprosjektet er å undersøke hvordan fysikkstudenters syn på programmering i fysikk utvikler seg i løpet av studiet og hva som motiverer dem.

Data fra dette prosjektet vil bli brukt i min masteroppgave ved institutt for fysikk.

Hvem er ansvarlig for masteroppgaven?

Henrik Storesund, femteårsstudent ved Teknisk fysikk, Institutt for fysikk, NTNU.

Hvorfor får du spørsmål om å delta?

Du får spørsmål om å delta fordi du går i første, andre eller tredje årstrinn på studiet fysikk og matematikk eller bachelor i fysikk ved NTNU. Jeg ønsker å intervju en og en student av gangen.

Hva innebærer det for deg å delta?

Du godtar å delta i et enkeltintervju. Intervjuet er ment å vare i omtrent 30 minutter. I intervjuet vil du få spørsmål som angår programmering i fysikk: din motivasjon og erfaring, dine syn på undervisningen. Jeg tar lydopptak og notater fra intervjuet, som lagres fram til prosjektets slutt, hvorpå datamaterialet vil bli anonymisert.

Det er frivillig å delta

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

Ditt personvern – hvordan vi oppbevarer og bruker dine opplysninger

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

- Det er kun oppgaveansvarlig Henrik Storesund som vil ha tilgang til personopplysninger, via koblingsnøkkel.
- Personopplysningene dine vil bli erstattet med en kode som lagres på en navneliste adskilt fra øvrige data (koblingsnøkkel). Datamaterialet vil lagres ved hjelp av en skybasert lagringsløsning.

I en eventuell publisasjon vil ingen utenforstående være i stand til å gjenkjenne deg. Dine personopplysninger vil ikke bli publisert.

Hva skjer med opplysningene dine når jeg avslutter masteroppgaven?

Masteroppgaven skal etter planen avsluttes 21.01.2022. Ved slutten av masteroppgaven vil dataene bli anonymisert, det vil si at koblingsnøkkel som gjør det mulig å kombinere dine personopplysninger og dine data vil bli slettet. Lydopptakene vil også bli slettet.

Dine rettigheter

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

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

Hva gir oss rett til å behandle personopplysninger om deg?

Vi behandler opplysninger om deg basert på ditt samtykke.

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

Hvor kan jeg finne ut mer?

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

- Student som gjennomfører masteroppgaven: Henrik Storesund, tlf: 476 46 124, e-post: hensto@stud.ntnu.no
- Personvernombud: Thomas Helgesen.
- NSD – Norsk senter for forskningsdata AS, på epost (personverntjenester@nsd.no) eller telefon: 55 58 21 17.

Med vennlig hilsen

Oppgaveansvarlig
(Student)

Eventuelt student

Samtykkeerklæring

Jeg har mottatt og forstått informasjon om masteroppgaven **Beregningsorientering i fysikkutdanningen**, og har fått anledning til å stille spørsmål. Jeg samtykker til:

- å delta i gruppeintervju

Jeg samtykker til at mine opplysninger behandles i masteroppgave frem til 21.01.2022.

(Signert av prosjektdeltaker, dato)

Vedlegg D: Meldeskjema

<https://meldeskjema.nsd.no/eksport/600a95ee-746d-4fa2-a47e-7339d025168f>

Referansenummer

906615

Hvilke personopplysninger skal du behandle?

Navn (også ved signatur/samtykke)

E-postadresse, IP-adresse eller annen nettidentifikator

Lydopptak av personer

Andre opplysninger som vil kunne identifisere en fysisk person

Type opplysninger

Du har svart ja til at du behandler andre opplysninger som vil kunne identifisere en person, beskriv hvilke

Personene som vil bli spurt om å delta, vil bli valgt ut på bakgrunn av det studieprogrammet de går på. Denne opplysningen vil det være relevant å nevne i en avsluttende rapport.

Skal du behandle særlige kategorier personopplysninger eller personopplysninger om straffedommer eller lovovertridelser?

Nei

Prosjektinformasjon

Prosjekttittel

Beregningsorientering i fysikkutdanningen

Prosjektbeskrivelse

Prosjektet omhandler beregningsorientering i fysikkemner på universitetsnivå. Bakgrunnen for prosjektet er at programmering representerer en forholdsvis ny ferdighet innen grunnleggende fysikkutdanning, selv om de aller fleste fysikere og ingeniører har benyttet programmering og beregningsorientering som verktøy og innfallsvinkel i mange år. Dette prosjektet skal undersøke om undervisningen og læringsaktivitetene som tilbys innen beregningsorientering er tilpasset studentenes ferdigheter. Gjennom kvalitative gruppeintervjuer med studenter på studieprogram for fysikk og matematikk ved NTNU, ønsker vi å finne et svar på dette. Intervjuene vil omhandle studentenes erfaring med programmeringsoppgaver fra emner de har gjennomført. Analyse av disse intervjuene vil gi et inntrykk av studentenes opplevelser av læringsaktiviteter knyttet til beregningsorientering, sett i lys av anbefalt rammeverk for beregningsorientering i fysikkundervisning.

Begrunn behovet for å behandle personopplysningene

Behovet er av ren praktisk art: det er nødvendig å innhente navn og e-post for å kunne avtale intervjuer med de aktuelle studentene. Den indirekte personopplysningen om studieprogram vil være relevant med tanke på at det eksisterer flere studieprogram ved NTNU med en klar fysikkprofil. Det er dermed behov for å klargjøre hvilke studenter som inngår i studien med tanke på den opplevde relevansen av beregningsorientering.

Ekstern finansiering

Type prosjekt

Studentprosjekt, masterstudium

Kontaktinformasjon, student

Henrik Storesund, hensto@stud.ntnu.no, tlf: 47646124

Behandlingsansvar

Behandlingsansvarlig institusjon

Norges teknisk-naturvitenskapelige universitet / Fakultet for naturvitenskap (NV) / Institutt for fysikk

Prosjektansvarlig (vitenskapelig ansatt/veileder eller stipendiat)

Magnus Strøm Kahrs, magnus.s.kahrs@ntnu.no, tlf: 99159606

Skal behandlingsansvaret deles med andre institusjoner (felles behandlingsansvarlige)?

Nei

Utvalg 1

Beskriv utvalget

Studenter ved Fysikk og Matematikk femårig masterstudium ved NTNU.

Rekruttering eller trekking av utvalget

Rekruttering vil basere seg på informasjon om prosjektet og invitasjon til deltagelse overfor samtlige studenter fra 1. til 3. studieår på det aktuelle studieprogrammet. Utvalget vil basere seg på frivillig, informert deltagelse.

Alder

19 - 25

Inngår det voksne (18 år +) i utvalget som ikke kan samtykke selv?

Nei

Personopplysninger for utvalg 1

Navn (også ved signatur/samtykke)

E-postadresse, IP-adresse eller annen nettidentifikator

Lydopptak av personer

Andre opplysninger som vil kunne identifisere en fysisk person

Hvordan samler du inn data fra utvalg 1?

Gruppeintervju

Vedlegg

Intervjuguide FINAL.docx

Grunnlag for å behandle alminnelige kategorier av personopplysninger

Samtykke (art. 6 nr. 1 bokstav a)

Informasjon for utvalg 1

Informerer du utvalget om behandlingen av opplysningene?

Ja

Hvordan?

Skriftlig informasjon (papir eller elektronisk)

Informasjonsskriv

Samtykkeerklæring FINAL.doc

Tredjepersoner

Skal du behandle personopplysninger om tredjepersoner?

Nei

Dokumentasjon

Hvordan dokumenteres samtykkene?

Manuelt (papir)

Hvordan kan samtykket trekkes tilbake?

Samtykket kan trekkes tilbake ved å kontakte prosjektansvarlig eller studenten som gjennomfører prosjektet (kontaktinformasjon til begge disse framgår av samtykkeerklæringen).

Hvordan kan de registrerte få innsyn, rettet eller slettet opplysninger om seg selv?

Ved å kontakte prosjektansvarlig eller studenten som gjennomfører prosjektet.

Totalt antall registrerte i prosjektet

1-99

Tillatelser

Skal du innhente følgende godkjenninger eller tillatelser for prosjektet?

Behandling

Hvor behandles opplysningene?

Ekstern tjeneste eller nettverk (databehandler)

Fysisk isolert maskinvare tilhørende behandlingsansvarlig institusjon

Hvem behandler/har tilgang til opplysningene?

Student (studentprosjekt)

Databehandler

Hvilken databehandler har tilgang til opplysningene?

Koblingsnøkkelen og transkripsjonene vil lagres i NTNUs skybaserte lagringsløsning, Onedrive for Business. Lydopptakene fra gruppeintervjuene vil lagres på en fysisk isolert maskinvare (altså IKKE i Onedrive).

Tilgjengeliggjøres opplysningene utenfor EU/EØS til en tredjestat eller internasjonal organisasjon?

Nei

Sikkerhet

Oppbevares personopplysningene atskilt fra øvrige data (koblingsnøkkel)?

Ja

Hvilke tekniske og fysiske tiltak sikrer personopplysningene?

Flerfaktorautentisering

Varighet

Prosjektperiode

22.01.2021 - 01.04.2022

Skal data med personopplysninger oppbevares utover prosjektperioden?

Nei, data vil bli oppbevart uten personopplysninger (anonymisering)

Hvilke anonymiseringstiltak vil bli foretatt?

Koblingsnøkkelen slettes

Lyd- eller bildeopptak slettes

Vil de registrerte kunne identifiseres (direkte eller indirekte) i oppgave/avhandling/øvrige publikasjoner fra prosjektet?

Nei

Tilleggsopplysninger

Vedlegg E: Temaer og koder fra intervjuene

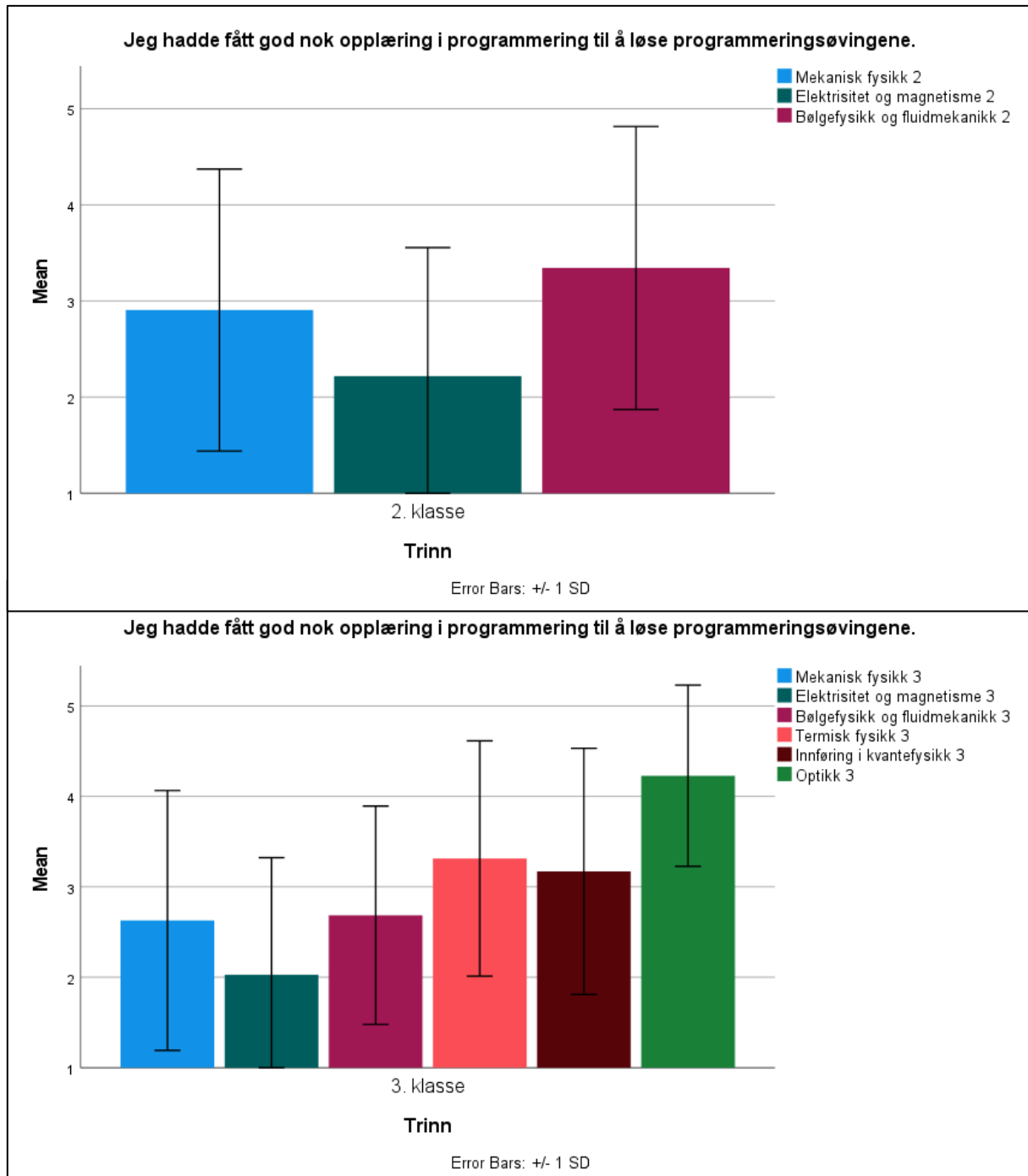
Tema	Undertema	Kode
Forhåndskunnskaper og -forventninger		Forhåndskunnskaper Forventet ikke at studiene skulle inneholde så mye programmering
Motivasjon	Mestringsforventning	Blir vanskeligere og vanskeligere å forstå fysikken Feilhåndteringsprosessen Kompenserer for manglende fysikkforståelse med å kunne programmeringen Uklare forventninger Uklare oppgaver Utfordringer knyttet til gruppearbeid Utfordringer når både fysikken og programmeringen er vanskelig Problemløsningsoppgaver oppleves nødvendigvis som litt krevende programmering blir mer overkommelig hvis man jobber med det jevnlig Programmeringsoppgaver mangler gradvis nivåøkning sammenlignet med vanlige oppgaver Programmeringsoppgaver og regneoppgaver er ganske like på et overordnet nivå Vanskelighetsgrad i programmeringsoppgaver
	Interesseverdi	Interessen for vanlige regneøvinger fra, mens interessen for programmering forble høy Mister lysten til å programmere pga å ikke få til oppgaver Programmering er gøy når... Sommerjobb med programmering gjorde det mer interessant
	Nytteverdi	Det at programmering innføres i videregående skole gir et bilde av programmering som mer nyttig Innså nytten av beregningsorienterting i fysikk først i andre semester Manglende programmeringskompetanse kommer i veien for det fysiske utbyttet Motivasjon for programmering er at det gir mestringsfølelse når det fungerer

		<p>Motivasjon for programmering er at det kan ha og bruke data slik en selv vil</p> <p>Motivasjon for programmering er forståelsen av at det er nyttig i arbeidslivet</p> <p>Nyttig fordi datamaskiner regner mye fortere enn en kan gjøre selv</p> <p>Nyttig fordi det kan hjelpe for å forstå fysikk</p> <p>Nyttig fordi man blir god på problemløsning</p> <p>Nyttig fordi man lærer å tenke logisk</p> <p>Nyttig pga det gir bedre innsikt i hvordan det ville vært in real life</p> <p>Nyttig pga fremtidig jobb</p> <p>Nyttig pga stor frihet</p> <p>Vil bare få programmeringsoppgaver gjort unna for å kunne ta eksamen</p>
	Kostnad	Arbeidsmengde i programmeringsoppgaver sammenlignet med vanlige oppgaver
Progresjon		Forskjellen mellom begynnelsen av studiet og nå
Systemet rundt studentene		<p>Foreleserne får et feilaktig inntrykk av studentenes nivå siden kun de flinkeste er aktive i forelesningene</p> <p>Forslag til forbedringer</p> <p>Får løsningen på programmeringsoppgavene fra medstudenter</p> <p>Hjelp ved å bruke utdelt eksempelkode</p> <p>Hjelp ved å spørre studentassistenter</p> <p>Hjelp ved å søke på nettet</p> <p>Ingen tilbakemelding på programmeringsoppgaver</p> <p>Programmering trenger ikke å nevnes i studiebeskrivelsen siden det er i ferd med å bli en selvfølge at det er med</p> <p>Programmeringsemnenes bidrag</p> <p>Savner undervisning om programmering i fysikkemnene</p> <p>Umulig å vite noe om et studie på forhånd</p>
Kjennetegn på en god programmeringsoppgave		<p>En god oppgave bidrar til å gjøre fysikken mer forståelig</p> <p>En god oppgave er delt inn i bolker</p> <p>En god oppgave gir tips for å komme i gang siden det er vanskelig å koble sammen programmeringen og fysikken</p> <p>En god oppgave inneholder informasjon om hvordan svaret skal plottes</p> <p>En god oppgave kan gjøres uavhengig av om man har løst den forrige oppgaven</p> <p>En god oppgave kan relateres til fysikkpensum ellers i faget</p> <p>Relativt dårlige læringsopplevelser i bestemte emner</p>

		Relativt gode læringsopplevelser i bestemte emner
--	--	---

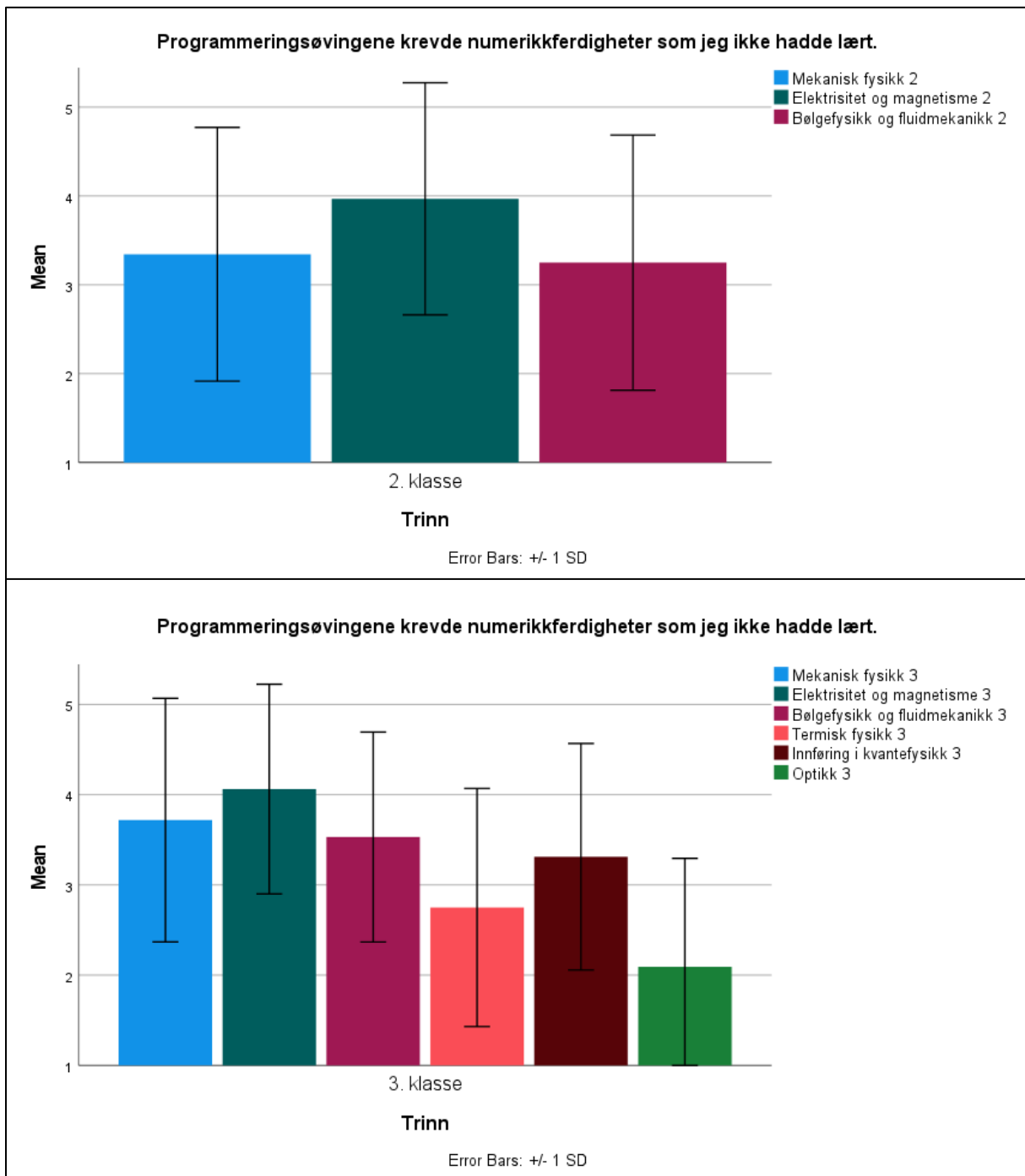
Vedlegg F:

Vedlegg F1



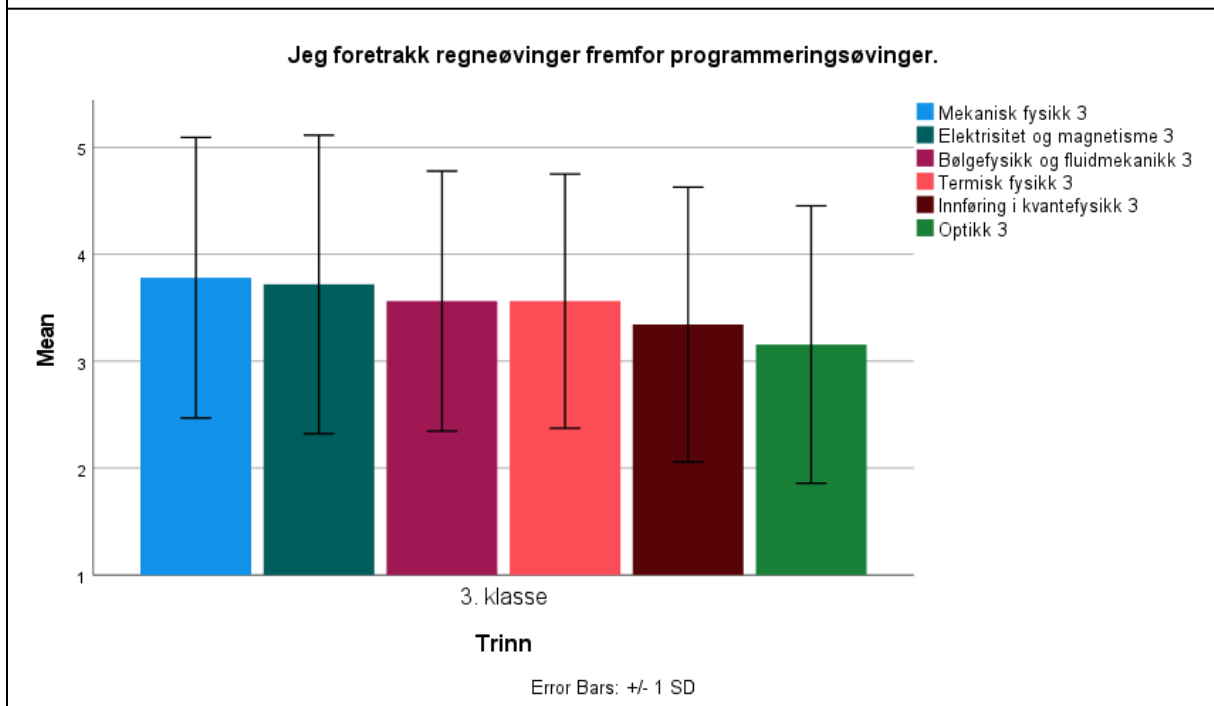
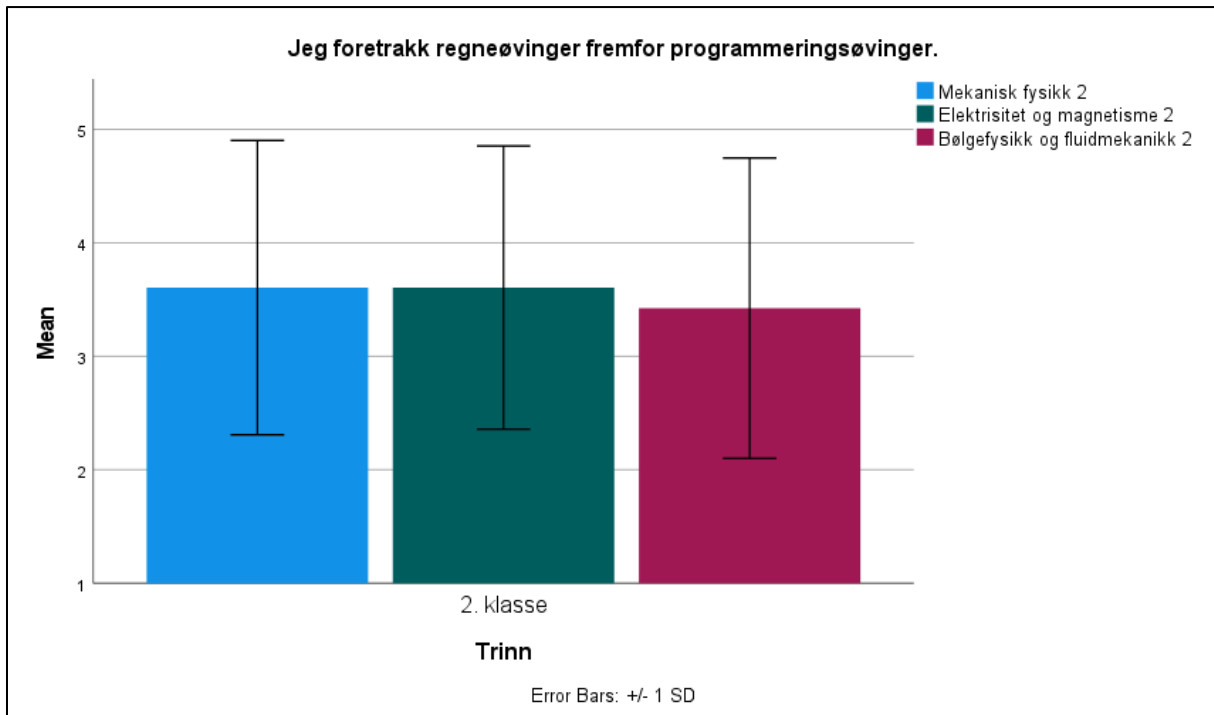
Vedlegg F1: Svar på påstand 3.1 for hvert enkelt emne besvart av studenter i andre og tredje årstrinn.

Vedlegg F2



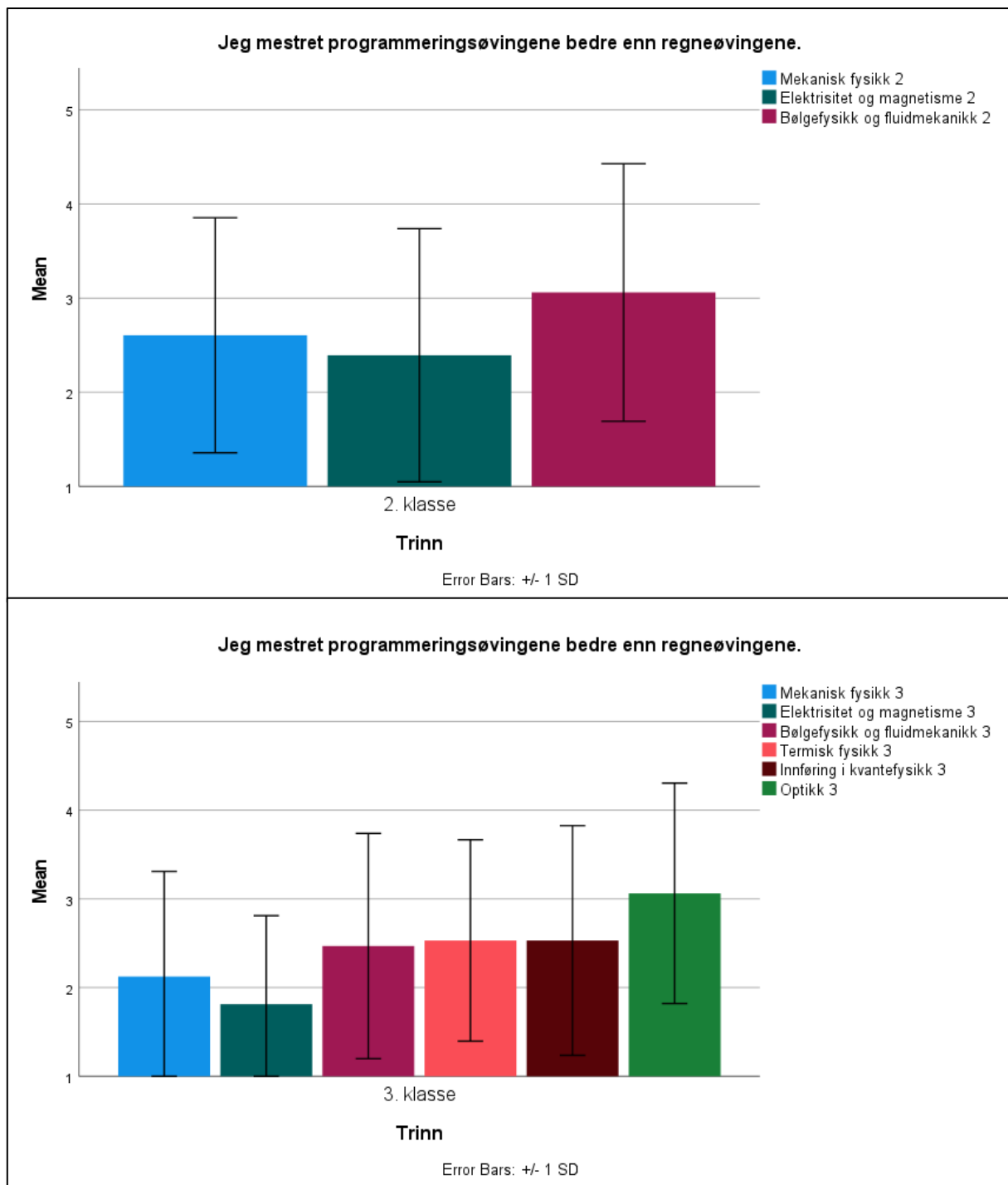
Vedlegg F2: Svar på påstand 3.2 per emne i andre og tredje årstrinn.

Vedlegg F3



Vedlegg F3: Svar på påstand 3.3 per emne i andre og tredje årstrinn.

Vedlegg F4



Vedlegg F4: Svar på påstand 3.4 per emne i andre og tredje årstrinn.