

Generiske ferdigheter «Arbeidsgruppe programmeringsemne»

Generiske ferdigheter «Arbeidsgruppe programmeringsemne»	1
Mandat	3
Om utvalget	3
Hovedkonklusjoner	3
Bakgrunn	4
Valg av programmeringsspråk	5
Undervisningsopplegg	6
Samkjøring mot andre kurs	7
Videre bruk av programmering	7
Mangler ved dagens pensum	8
Student volum	8
Oppsummering fra møter med programutvalg	8

Mandat

Utvalget ble gitt følgende mandat:

«Arbeidsgruppens anbefaling er at alle studenter ved fakultetet skal ha et obligatorisk emne i programmering i løpet av første studieår. Emnet gis i regi av Institutt for informatikk, men ulike fagnære eksempler og problemstillinger integreres i samarbeid med de ulike studieprogrammene. Gruppen skal utforme et forslag til hvordan et slikt emne bør utformes med hensyn til innhold og struktur, hvilke behov brukerinstuttene har for lærerstøtte (bl.a. når programmeringskunnskapene skal anvendes i videre emner) og hvordan det kan organiseres for å best mulig integrere emnet i første studieår for alle studenter.»

Om utvalget

Utvalget har bestått av:

Fredrik Manne, Institutt for informatikk, leder
Mo Yan Yuen, Institutt for informatikk, sekretær for gruppen
Øyvind Fiksen, Institutt for biovitenskap
Bjarte Hannisdal, Institutt for geovitenskap
Any Helene Bagge, Institutt for informatikk

Utvalget har avholdt 9 møter, og har møtt alle undervisningsutvalg ved MN med unntak av Geofysisk institutt. GFI har imidlertid sendt brev med sine synspunkter til utvalget. Det har også vært et eget møte med arbeidsgruppen som ser på «Nytt brukerkurs i matematikk».

For å danne grunnlag for diskusjon ble følgende spørsmål sendt til programstyrene i forkant av møtene:

- 1) Hva forventer dere at et obligatorisk programmeringsemne skal inneholde?
- 2) Hvordan vil dere integrere dette emnet i det videre studieløpet?
- 3) I hvilket semester ønsker dere at det tilbys for 1. årstudenter: 1. semester (høst) eller 2. semester (vår)?

Hovedkonklusjoner

- Stort sett alle instituttene ser det som positivt at programmering blir obligatorisk for alle 1. års bachelorstudenter. Et slikt kurs bør også lære studentene algoritmisk tenkning.

- Man ønsker at alle studentene skal ha samme pensum og at man ikke differensierer læringsinnhold avhengig av hvilket studieprogram en student tilhører.
- Det er bred enighet om å bruke Python som programmeringsspråk. Det er likevel ønskelig at man orienterer om andre relevante programmeringsspråk som studentene kan støte på senere i studiet.
- Øvingsoppgaver bør i størst mulig utstrekning være faglig relevante i forhold til det enkelte studieprogram som studentene følger.
- Undervisningsopplegget og fremdrift bør fastlegges i forkant av hvert semester slik at det er mulig for kurs som undervises parallelt å forutsette hva som har blitt undervist i INF100 til en hver tid. På denne måten vil en muliggjøre bruk av programmeringseksempler i disse kursene. Det foreslås derfor at det blir opprettet en gruppe med fagansvarlige for relevante innføringsemner. Denne skal møte minst en gang i forkant av hvert semester for å gå igjennom undervisningsplanen til INF100.
- De fleste instituttene ønsker å tilby INF100 i 1. semester (høst).

Bakgrunn

Institutt for informatikk hadde frem til høsten 2018 to kurs i grunnleggende programmering: INF100 «Innføring i programmering», samt et brukerkurs i programmering: INF109 «Dataprogrammering for naturvitenskap». INF100 var opprinnelig rettet mot informatikk studenter og brukte Java som programmeringsspråk, mens INF109 var tiltenkt andre realfagstudenter ved det Matematisk-naturvitenskapelige fakultet og brukte Python som programmeringsspråk. Emnet INF109 ble sist undervist våren 2018 og deretter erstattet med INF100 fra høsten samme år. Det medfører at INF100 nå er det eneste innføringsemnet i programmering ved MN. Samtidig med omleggingen fikk INF100 ny emnebeskrivelse og kurset er nå i større grad modellert etter det gamle INF109, bla bruker kurset Python som programmeringsspråk.

Våren 2019 bruker INF100 boken «Practical Programming, 3rd ed, An Introduction to Computer Science Using Python 3.6» av Paul Gries, Jennifer Campbell og Jason Montojo. Se <https://www.uib.no/emne/INF100> for mere informasjon om kurset og <https://mitt.uib.no/courses/16312> på mittuib.no (krever innlogging).

Kurset undervises over 17 uker, hvorav siste uke er satt av til repetisjon. Pensum består av de 15 første kapitlene i læreboken (unntatt kap. 12) og dekker i hovedsak følgende tema:

- introduksjon til programmering,
- lage og bruke funksjoner,
- arbeide med tekst,

- betingelser,
- importere moduler,
- metoder,
- datastrukturer (samlinger og lister),
- løkker,
- lese og skrive til filer samt lagring av data som samlinger,
- søking og sortering,
- objektorientert programmering,
- testing og feilsøking.

INF100 er i dag obligatorisk eller sterkt anbefalt i en rekke studieretninger ved MN. Høsten 2018 var 447 studenter oppmeldt til eksamen, hvorav 392 møtte. Våren 2019 er det 155 som er meldt til eksamen. Kurset er også obligatorisk i noen studieprogram utenfor MN, så som bachelor programmene i Informasjons- og kommunikasjonsteknologi (IKT) og Kognitiv Vitenskap (KogVit) som begge organiseres fra SV fakultetet.

Det er helt klart at et obligatorisk kurs i programmering for alle bachelor studenter ved MN vil bygge på en videreføring av INF100.

Valg av programmeringsspråk

Både i undervisning og i forskning brukes det flere ulike programmeringsspråk ved MN. Ved Institutt for biovitenskap og på Matematisk institutt bruker man bla programmeringsspråket R som er spesielt egnet for statistiske beregninger. Flere institutt bruker Matlab og IFT bruker også Fortran. I tillegg er det utstrakt bruk av regneark i flere studier.

Medlemmene i utvalget mente i utgangspunktet at det ville være fordelaktig å gjennomføre all undervisning med et felles programmeringsspråk og da fortrinnsvis Python. Internasjonalt har Python i stor grad overtatt for Java som første programmeringsspråk i høyere utdanning. Det er flere grunner til dette, men først og fremst at det er lett å komme i gang og at det er enkelt å bruke til grunnleggende databehandling.

Ved Universitetet i Oslo bruker man også Python som opplæringspråk for alle MN studenter, men her deler man opp undervisningen mot slutten av kurset slik at studenter får en kort introduksjon til ulike programmeringsspråk tilpasset sitt studieprogram. Man har også et eget undervisningsopplegg for bio-studenter.

På direkte spørsmål har samtlige undervisningsutvalg vært positive til at man bruker Python som programmeringsspråk. Det har blant annet blitt pekt på at det finnes gode programbibliotek innenfor ulike fagområder (f.eks NumPy for vitenskaplige beregninger). Flere har også sagt at man ønsker at alle studentene bør få den samme undervisningen og at man ikke bør differensiere innhold for de ulike studieprogrammene. Hovedargumentet for dette er at man da på senere kurs som taes av studenter fra ulike studieprogram, kan gjøre forutsetninger om hvilken forkunnskap studentene har. Det er likevel ytret ønske om at studentene får en grunnleggende forståelse for hvilke andre programmeringsspråk de kan forvente å møte senere i studiet og hovedforskjellene mellom Python og disse.

Undervisningsopplegg

For å lære seg å programmere er det nødvendig å bruke en god del tid på å løse oppgaver. I INF109 var det opprinnelig meningen at oppgavene skulle bestå av problemstillinger motivert fra ulike fagområder ved MN. Over tid har dette imidlertid sklidd noe ut og øvingsopplegget har hatt mindre forankring i andre MN miljø. I det nye INF100 kurset har man igjen henvendt seg til andre fagmiljø og bedt om innspill til oppgaver. Slike oppgaver har måttet tilpasses slik at de har kunnet bli løst av samtlige studenter på kurset. Dette medfører blant annet at man ikke kan forutsette spesifikk kunnskap om et bestemt fagområde.

Flere programstyrer har understreket at det er viktig at studentene må se relevansen av det de lærer. Det har blitt kommentert at studenter har opplevd øvingsopplegget som fjernt fra sine studier og ofte lagt opp mot mer rene «data» problemstillinger, som for eksempel spillutvikling.

For at kurset skal føles relevant for studentene bør oppgaver så langt som mulig være koblet mot studieprogrammet de følger. Det er derfor ønskelig at det blir utarbeidet egne oppgaver for hvert studieprogram. Dette vil kreve innsats fra de ulike fagmiljøene, men over tid vil man opparbeide tilstrekkelig med oppgaver til at man får en «pool» som kan gjenbrukes. Denne typen oppgaver kan i mindre grad brukes i plenumsforelesninger, og vil måtte håndteres i individuelle grupper for de ulike studieprogrammene. Dette vil kreve at gruppelederne har nødvendig fag-kunnskap, både innen det bestemte faget og i programmering.

I den grad oppgaver skal være tellende for endelig karakter, må de samkjøres slik at de har en enhetlig vanskelighetsgrad. Oppgavene må også utvikles slik at de passer å bli gitt på samme tid basert på hvor langt man har kommet i kurset. Det er også et poeng at siden oppgavene blir fagspesifikke så blir det også vanskelig å bruke de til eksamen som forventes å være felles for alle studenter.

Når man utvikler oppgaver er det vesentlig at man tar hensyn til at kurset skal taes av alle studenter ved MN og det vil være et av de første kursene de møter. Det er derfor viktig å ikke bli overambisiøs, men heller sørge for at studentene opplever mestring.

Følgende gir ulike overordnede tema som har vært nevnt for slike oppgaver:

- Bør arbeide med store datamengder slik at man ser hvilke muligheter databehandling (problem bør ikke kunne løses for hånd).
- Praktisk tilnærming.
- Lære seg å bruke eksisterende kode.
- Feilsøking
- Plotting av data

Samkjøring mot andre kurs

En av synergiene man kan oppnå ved at alle studenter ved MN tar kurs i programmering er at andre kurs som undervises parallelt med INF100 også kan bruke eksempler og oppgaver som forutsetter kunnskap om programmering. Det vil gjøre det mulig å foreta simuleringer der man ellers tidligere bare har kunnet arbeide analytisk. Denne muligheten har vært diskutert med de ulike programstyrene og flere har vært positive. For å gjennomføre en slik ordning må fremdriften i INF100 være fastlagt i god tid før semesterstart.

Selve gjennomføringen forutsetter at man først gjør en vurdering hvilke kurs det er aktuelt å samkjøre med. Undervisere for de aktuelle kursene må så komme sammen med den eller de som underviser INF100 for å planlegge fremdrift. Her kan man også se på mulige fagspesifikke øvingsoppgaver som nevnt ovenfor. For at opplegget skal foregå på en systematisk måte bør et slikt møte innkalles av studieadministrasjonen ved Institutt for informatikk.

Det forutsettes at undervisere ved aktuelle kurs har nødvendig kompetanse i Python-programmering. Dette er ikke alltid tilfelle, og det har derfor blitt ytret ønske om at man tilbyr kurs i Python-programmering for ansatte. Administrasjon og gjennomføring av et slikt kurs bør kunne håndteres av Institutt for informatikk. Kurset vil også være aktuelt for andre ansatte ved MN som ønsker å bruke programmering i undervisningen.

For å starte opp en ordning der man synkroniserer undervisningen i begynnerkursene har Det pedagogiske akademi ved MN sagt seg villig til å stille med midler slik at man kan gjennomføre en todagers samling for de berørte miljøene høsten 2019.

Videre bruk av programmering

Samtlige undervisningsutvalg har blitt bedt om å vurdere hvordan de kan integrere programmering i sin videre utdanning. Hensikten med dette er å unngå at INF100 skal oppleves som et frittstående kurs og å sikre at studentene skal vedlikeholde kunnskapen de har tilegnet seg. Ansvar for at en slik integrering blir foretatt ligger på de enkelte studieprogrammene, men undervisningsutvalget ved MN må kontrollere og følge opp.

Mangler ved dagens pensum

Av tema som har vært etterlyst men som ikke dekkes fullt ut i dagens pensum kan nevnes:

- plotting av data,
- bruk av eksterne bibliotek/kode,
- algoritmisk tenkning,
- introduksjon til andre programmeringsspråk.

Det kan tenkes at noe av dette kan erstatte «objektorientert programmering» og «søking og sortering». Til sammen dekker disse temaene tre uker med undervisning. Begge disse temaene er mer rettet mot studenter som skal studere data videre. Det burde muliggjøre at man flytter de opp i neste programmeringskurs (for tiden INF101).

Student volum

Høsten 2018 var det totalt ca 655 startende studenter ved MN, IKT og Kognitiv vitenskap (de to siste administreres av SV fakultetet, men tar INF100). Det var 447 studenter som meldte seg til eksamen høsten 2018 og 155 til våren 2019. Det er grunn til å tro at eksamensmeldte studenter også inkluderer studenter som tar emnet om igjen. Hvis INF100 gjøres obligatorisk må det derfor forventes en ikke ubetydelig økning i antall studenter. Det er derfor ønskelig å få til en best mulig fordeling av studentene mellom høst og vår semesteret. Så langt har Biovitenskap og Geovitenskap sagt at vårsemesteret passer for deres studenter, de resterende ønsker høstsemesteret. Tilsammen utgjorde dette 185 studenter høstsemesteret 2018 (Bachelor i biologi, molekylærbiologi og geovitenskap).

Hovedårsaken til at så mange ønsker høstsemesteret er at INF100 er tenkt å erstatte Ex.Phil som flyttes til senere i studiet. Det vil være en utfordring å håndtere et større antall studenter i høstsemesteret enn man allerede gjør i dag. Det er derfor ønskelig at man går en runde til for å se om det er muligheter for at flere studieprogram tar INF100 om våren

Oppsummering fra møter med programutvalg

Institutt	Ønsket semester	Språk	Forventinger/ ønsker/ viktig
KI	Høst	Python	<ul style="list-style-type: none"> ▪ Relevant faglig innhold ▪ Utfordring med farmasi (og andre profesjonsstudier ved MN): lite rom for endringer i studieplanen ▪ Algoritmisk tenking ▪ Kunnskap å lære nye programmeringsspråk etter endt kurs ▪ Ikke bli overambisiøse: studenter må ha en mestringsfølelse etter endt kurs ▪ Ikke gjør programmeringsemnet obligatorisk fordi man må; det må være en kvalitetsløft
MI	Høst	Python	<ul style="list-style-type: none"> ▪ Algoritmisk tenking ▪ Husk lærerutdanning: programmering blir viktig. Ønsker derfor algoritmisk tenking i læringsutbytte (enten i form av eksempler i forelesninger eller i oppgaver)
BIOV	Høst, eller vår, dersom ex.phil flyttes ut av 2. semester i første året	Python/R	<ul style="list-style-type: none"> ▪ Bruk av faglig relevant innhold (BIO, MOL og fellesgrad) – i form av oppgaver ▪ BIOV må se på hvordan programmeringskunnskapene integreres videre i studieløpet: hvert semester eller blir det i form av valgemner? ▪ Ikke alle vitenskapelige ansatte ved BIOV kan programmering. Hvordan kan II bidra? Kanskje Bio-INF ansatte og stipendiater kan hjelpe med undervisning.
GEOV	Vår	Python?/ MATLAB?	<ul style="list-style-type: none"> ▪ Faglig relevant innhold ▪ Ikke gjør det for vanskelig for studentene. ▪ Vise relevansen/verdien av programmering for geologene.
IFT	Høst	Python	<ul style="list-style-type: none"> ▪ Relevant faglig innhold (oppgaver knyttet til fagområdet) ▪ Algoritmisk tenking ▪ Beregninger
GFI	Høst	Python	<ul style="list-style-type: none"> ▪ Relevant faglig innhold ▪ GFI ønsker kurs i Python for de vitenskapelige ansatte og hjelp til å «oversette» øvelsene fra andre program til Python. ▪ Læringsutbytte: Bruk av eksisterende kode/programmer, Endring av eksisterende kode, Skrive egne programmer, Strategier til feilsøking, Innlesing av data i ASCII og netCDF format, Plotting av data, Enkel dataanalyse, som gjennomsnitt, standardavvik, regresjon, mm, vektorer og matriser (2D, 3D)