

Kort om IN1900, IN-KJM1900 og IN-GEO1900

Joakim Sundnes¹ Ole Christian Lingjærde²

Simula Research Laboratory and Department of Informatics, University of Oslo¹

Department of Informatics, University of Oslo²

Aug 18, 2026

Velkommen til IN1900, IN-KJM1900 og IN-GEO1900!



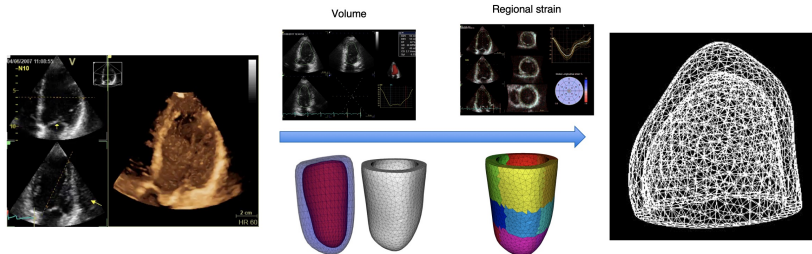
© Scott Adams, Inc./Dist. by UFS, Inc.

- Tirsdag 18. august:
 - Generell info om IN1900, IN-KJM1900, IN-GEO1900
 - Kort gjennomgang av info på semestersidene
 - Nyttige tips om programvare og arbeidsform
- Torsdag 20. august:
 - Live-programmering, intro til terminalvindu, etc.
 - Programmering av matematiske formler (kap 1 og 2)

- Alle forelesninger vil bli tatt opp og lagt ut på nett.
- Tilgang til opptak krever innlogging
- Eventuelle spørsmål fra salen vil komme med i opptaket, men kan redigeres bort hvis det er ønskelig.

- Master (Cand Scient) i matematikk fra UiO (1998)
- PhD i Informatikk fra UiO (2002)
- Forsker på Simula/UiO
- Undervist IN1900 siden 2015

Forskning; matematiske modeller av hjertet



- Fysikkbaserte modeller for elektrofysiologi og mekanikk
- Systemer av differensiallikninger - umulig å løse analytisk
- Løses med ulike numeriske metoder - programmert i C++ og Python

- Startet i 2007 av Hans Petter Langtangen
- Nytt og revolusjonerende da det kom:
 - Python
 - Spesielt tilpasset naturvitenskap
- Introduksjonskurs:
 - Forutsetter **ingen** forkunnskap om programmering
 - Varierende utgangspunkt (som i alle introkurs i programmering)

Programmeringsfaget er i endring

- IN1900/INF1100 nesten uforandret fra 2007-2022
- Noen kunne programmere fra før, de fleste ikke
- To store endringer de siste årene:
 - Programmering introdusert på videregående skole
 - Generativ AI er *vel*dig god til å programmere!

Programmering i VGS (1)

- Flere og flere har programmert (litt el mye) tidligere:
 - 2018: Ca 25%
 - 2023: Ca 65%
 - 2024: Ca 80%
 - 2025: Ca 88% (60% "litt", 28% "mye")

Python bakgrunnssjekk



Programmering i VGS (2)

Fra læreplanen for Matematikk R2, kompetansemål:

- *Utforske rekursive sammenhenger ved å bruke programmering og presentere egne framgangsmåter*
- *Utvikle algoritmer for å beregne integraler numerisk, og bruke programmering til å utføre algoritmene*
- *Gi eksempler på ulike situasjoner som kan modelleres ved å bruke ulike matematiske funksjoner, og modellere og analysere slike situasjoner ved å bruke reelle datasett*

Erfaringer til nå:

- Store variasjoner mellom skoler/lærere
- Mange avanserte tema, men lite fokus på helt grunnleggende ting
- IN1900 => Like store nivåvariasjoner som tidligere

Resultater fra forkunnskapstest (1)

- Arrangert under forkurs i informatikk forrige uke
- Ca 160 svar (ikke helt representativt utvalg)
 - Eksamensår 2024-2026: 108
 - Matematikk R2: 115
- Fire tema:
 - Tilordning (Assignment)
 - Lister
 - Løkker
 - Funksjoner
- Varierende resultater, som forventet

Resultater fra forkunnskapstest (2)

Eksempler på resultater:

What does this program print?

```
a = 2
b = 5

a = a + b
b = a + b

print(a)
print(b)
```

```
-----
N answers      : 158
Correct        : 66 (41.77%)
Incorrect      : 69 (43.67%)
I don't know   : 23 (14.56%)
Did not answer: 0
```

False and False

```
-----
N answers      : 156
Correct        : 59 (37.82%)
Incorrect      : 50 (32.05%)
I don't know   : 47 (30.13%)
Did not answer: 2
```

Top 3 most common answers:

```
false  59
true    50
```

What does this program print?

```
ls = [10, 20, 30]
print(ls[1])
```

```
-----
N answers      : 156
Correct        : 39 (25.0%)
Incorrect      : 82 (52.56%)
I don't know   : 35 (22.44%)
Did not answer: 2
```

Eksempler på kommentarer:

- "Har jobbet med hva man skal skrive inn og ikke hvordan det skrives ut, så ble litt for ulikt fra vgs"
- "mange oppgaver er for like, kartlegger ikke bredt nok"
- "Veldig fort gjort å gjøre slurvefeil ved hvordan listene printes ut"

Hvorfor lære programmering? (2007-versjon)

Jeg begynte på UiO for å lære Matematikk/Fysikk/Kjemi/etc.
Hvorfor må jeg lære å programmere?

- Programmering blir viktigere og viktigere i industri og forskning, og brukes i flere og flere fagfelt
- Programmering vil bli brukt i veldig mange senere emner
- Hvorfor? Programmeringen gjør matematikk mye mer nyttig (og enklere)
- Økt fokus på maskinlæring og "data science" det siste tiåret har gjort "vår" type programmering mer og mer relevant

Hvorfor lære programmering? (2026-versjon)

KI løser alle problemene lynraskt og gjør nesten aldri feil. Hvorfor må jeg lære å programmere?

Fra "The lost interview" med Steve Jobs, 1995:

<https://youtu.be/BRT0IPdyPYU?si=0bYgsXp2eHTI-KkA>

Generativ AI i programmering (1)

- Alle oppgaver i IN1900 løses enkelt av ChatGPT og andre verktøy
- Også ganske avanserte oppgaver lar seg vanligvis løse
- Alle programmerere bruker AI-verktøy daglig
- Status 2023-2025:
 - AI gjorde av og til feil
 - Nødvendig å kunne lese, forstå, og rette opp koden som genereres

Generativ AI i programmering (2)

- Status 2026:
 - Nye språkmodeller løser "alle" programmeringsoppgaver, nesten uten feil
 - "Alle" software-utviklere bruker KI
 - AI brukes mer og mer også til å sjekke og teste generert kode
- Mindre behov for å lese og sjekke koden
- Dette betyr **ikke** at det er unødvendig å kunne programmere selv
 - Fortsatt nødvendig å forstå hva man gjør
 - AI gjør fortsatt av og til feil (veldig sjelden)
 - Basiskunnskap er fortsatt nødvendig/nyttig for alle
 - Nyttig å kunne lese, forstå, og evt rette opp koden som genereres
- Programmering blir mer effektivt og morsommere, men krever fortsatt innsats og kunnskap
- Eksamen i IN1900 er uten hjelpemidler!

Hva lærer man i kursene

1 Generell programmering:

- Hvordan oversetter vi et matematisk problem til kode?
- Struktur på programmer og kode
- Generelle verktøy og begreper, felles for alle språk

2 Python-spesifikt:

- Python-syntaks
- Python-spesifikke funksjoner og programmeringsstil

3 Verktøy:

- Terminalvindu, VS code, etc.

IN-KJM1900 og IN-GEO1900 = IN1900 uke 1-6 (ca) + prosjekt

- Uke 34-40: Samme forelesninger og oppgaver som IN1900
 - forelesninger: uke 34-40
 - gruppeøvelser (oppgaver): uke 35-40
- Samme midtveiseksamen som IN1900
- Eget kjemi/geo-relatert programmeringsprosjekt etter det

- IN1900: <http://www.uio.no/studier/emner/matnat/ifi/IN1900/h26>
- IN-KJM1900: <http://www.uio.no/studier/emner/matnat/ifi/IN-KJM1900/h26>
- IN-GEO1900: <https://www.uio.no/studier/emner/matnat/ifi/IN-GEO1900/h26>
- Se spesielt **IN1900 undervisningsplan** for info om hva som skjer hver uke

Undervisningen består av øvelser og forelesninger

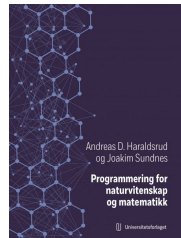
- Plenumsundervisning tirsdager 14.15-16 og torsdager 12.15-14
 - Oppgaveløsning (live-programmering)
 - Gjennomgang av nytt stoff (lysbilder, notebooks)
- 2 t oppgaveløsning på terminalstue i mindre grupper der du kan få individuell veiledning
- Anbefaling: Delta på *alle* undervisningstimene i starten av semesteret!
- Opptak av forelesningene blir lagt ut på emnesidene (etter ca 3-4 timer)

- Introduksjon av nye konsepter (lysbilder)
- Eksempler og oppgaver (live-programmering)
- Ca 2/3 live-programmering, 1/3 lysbilder
- Noe introduksjon av nytt stoff blir rent selvstudium
 - Morsommere og mer lærerike forelesninger
 - Større krav til forberedelser!

- Tidligere: Mest individuell jobbing med oppgaver, gruppelærer svarer på spørsmål ved behov
- Nå: KI svarer på alt, mye mindre behov for å stille spørsmål
 - Mer undervisning og opplegg i gruppetimene
 - Repetisjon, diskusjon, samarbeid,...

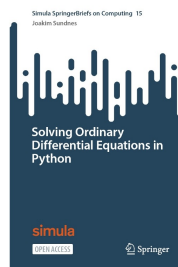
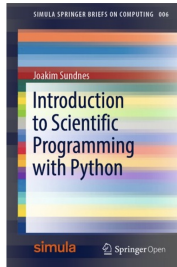
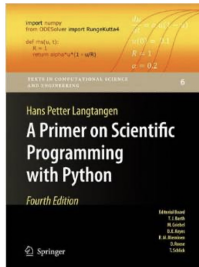
Undervisningsmaterieell (1)

- Ny lærebok fra høsten 2025
- Første gang med lærebok på norsk!
- Vi går gjennom ca ett kapittel pr uke, stort sett med ett til to nye sentrale begreper
- Ikke alle kapitler er pensum (liste kommer senere)



Undervisningsmaterieel (2)

- "A primer on ..." brukt fram til 2019, "Introduction to ..." 2020-2024.
- Spesielt "A primer on ..." av H. P. Langtangen er en nyttig tilleggsressurs, som har mye mer info og flere eksempler enn den nye læreboka
- Langtangen-boka er skrevet for Python 2, men forskjellene er små.



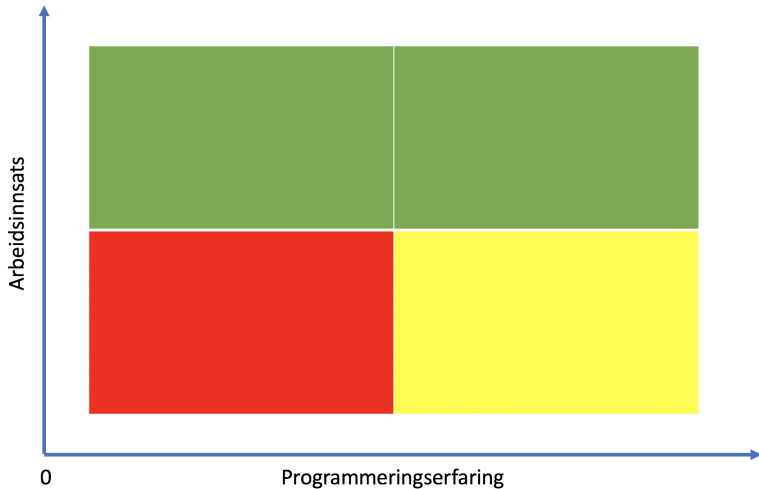
Python 2 vs Python 3

- Python 3 dominerer i dag
 - Python 3.14 er nyeste stabile versjon
- Lærebok, foiler, oppgaver i plenum etc er Python 3 (≥ 3.6)
- Gammel lærebok (Langtangen) er Python 2.7
- For IN1900 er forskjellen relativt liten (se mer info på emnesidene)

Kursene krever **ingen** forkunnskaper om programmering

- Vi starter med **veldig** enkle programmer, men går relativt fort fram
- For mange er programmeringen en helt ny måte å bruke en datamaskin på
- En del nye verktøy; editor, terminalvindu, etc
- Verktøy må læres underveis; kom tidlig i gang, spør om hjelp ved behov
- Har du aldri programmert før? Kom tidlig i gang, prioriter de første ukene.
- Har du programmert Python tidligere? Følg med på hva som skjer i undervisningen likevel

Mer om forkunnskaper og arbeidsinnsats



Siden 2023: færre obligatoriske oppgaver

- Tidligere versjon (2007-2022):
 - Obligatoriske innleveringer hver uke
 - Jevn jobbing gjennom semesteret -> veldig gode resultater
- 2023-2025:
 - Ukentlige oppgaver frivillig, men anbefales sterkt!
 - Oppgaver kan leveres inn, for retting og tilbakemelding fra gruppelærere
 - To obligatoriske oppgaver, den første i uke 36

- Midtveiseeksamen tirsdag 6. oktober
 - To timer (fire timer til i fjor)
 - Ren flervalgsprøve
 - Kodeforståelse og Python-syntaks
- Avsluttende fredag 11. desember
 - Fire timer (ingen endring)
 - Flervalg + små programmeringsoppgaver
- Ingen hjelpemidler

Anbefalt arbeidsform

- Minimum:

- Les ukens kapittel i læreboka
- Les om ukens læringsmål i undervisningsplanen, sørg for at du har fått med deg disse
- Forsikre deg om at du har kontroll på de sentrale begrepene.
- Før hver oppgaveløsning i plenum må du selv ha forsøkt å løse oppgavene
- Etterarbeid oppgavene når du har sett løsning i plenum
- Nå er du klar for ukens oppgaver, på egen hånd eller på gruppetime

Merk:

- Hvis du kan stoffet er det vanligvis mulig å gjøre alle ukens oppgaver i løpet av en gruppetime (2 timer)
- Å gå rett på oppgavene uten forberedelser ender ofte i mange timers prøving og feiling

Du må lære programmering ved å programmere mye

- Du kan ikke lese deg til programmering
- Mange synes programmering er krevende i begynnelsen - så blir det utrolig gøy!
- Oppskrift på suksess: vær godt forberedt til undervisningen - det gir deg mest fritid og mest læring
- Forventet arbeid er 13 timer med IN1900 hver uke (6 t undervisning, 7 t selvstudium)

Hvor mye matematikk må jeg kunne på forhånd?

- Nesten alle eksemplene i IN1900 handler om bruk av matematikk
- Vi bygger (i prinsippet) på R2 fra vgs
- Men matematikken i IN1900 er stort sett *numerisk* matematikk
- Vi håper at IN1900 skal belyse matematikk fra en ny vinkel og hjelpe deg til å forstå matematikk bedre samtidig som du lærer å programmere

Språk: miks av norsk og engelsk

- Muntlig undervisning foregår på norsk
- Ny lærebok på norsk
- Lysark på engelsk (stort sett)
- Nyttig å kunne både norske og engelske uttrykk!

Bruk av egen laptop, operativsystem, etc

- Man kan (i prinsippet) gjennomføre kurset uten egen laptop, alle oppgaver kan gjøres på terminalstuene
- Egen laptop anbefales; Python kjører på Windows, Mac og Linux
- Anbefalt arbeidsform; VSCode + evt terminalvindu
- Installasjon:
 - Python (3.1x) med pakkene NumPy og Matplotlib
 - VSCode (Visual Studio COde). Editor/IDE
- Se emnesidene for info om programvare og isntallasjon
- Spør AI om hjelp til installasjon
- Spør om hjelp på gruppetimene

Viktig info om UiO-brukernavn

- Du trenger UiO-brukernavn og passord for å:
 - Logge inn på maskiner på terminalstuene
 - Logge inn på UiO-maskiner hjemmefra
 - Levere oppgaver i [devilry](#)
- Din UiO-bruker er også en mail-adresse (`brukernavn@student.matnat.uio.no`). Sjekk denne jevnlig!

Bruk av AI-verktøy og kilder på web

- De fleste innleveringsoppgaver er frivillige, og man kan bruke AI-verktøy og kilder på nett så mye man ønsker
- Poenget med å levere er å få kommentarer og tilbakemelding fra gruppelærere => ikke noe poeng å levere AI-genererte løsninger
- Bruk AI for å lære, ikke for å løse oppgavene
- Eksamen er uten hjelpemidler!
- **Tips:**
 - Feilmeldinger fra Python kan limes rett inn i chatGPT e.l. (evt også Google søkefelt). Gir ofte en bra forklaring på hva som er galt.
 - Små programmer kan limes rett inn i ChatGPT med spørsmål som "hva er feil her?"
- Mulige "feller":
 - Veldig fristende å la AI generere en løsning som man deretter leser og "forstår"
 - Svar fra AI er ikke tilpasset IN1900 (men du kan be om det)
- Husk at eksamen er uten hjelpemidler

- Bruk AI-verktøy så mye du ønsker
- Du kan bli bedt om å forklare muntlig hva du har gjort.
- Noen oppgaver har spesielle krav til hvordan ting skal gjøres. Hvis løsningen ikke oppfyller kravene blir den ikke godkjent.
- Ikke kopier løsningsforslag fra nett direkte! Dette regnes som fusk, og man risikerer utestengelse fra UiO.

Samarbeid om obligatoriske oppgaver

- Samarbeid er bra, men pass på at alle bidrar, og at alle forstår de løsningene man kommer fram til
- Alle skriver sin egen kode, og leverer sin egen innlevering.
- Skriv en kommentar i innleveringen om hvem du har samarbeidet med:
 - Ved samarbeid blir koden gjerne veldig lik
 - IFI har verktøy som finner like innleveringer, og flagger dette som mulig avskrift/fusk
 - Veldig enkelt å rydde opp i hvis vi vet hvem som har samarbeidet med hvem. Tar fort mye tid ellers

Tilbakemelding på undervisningen

- Still spørsmål underveis
- Send epost med synspunkter etter forelesning
- Hva fungerer bra, hva er mindre bra?
- Er noen tema spesielt vanskelige?
- Er info på web utdatert eller feil?
- Alle innspill blir vurdert, selv om ikke alle forslag lar seg gjennomføre

Viktig hjemmearbeid første uke

- Les gjennom informasjon på semestersidene:
 - Sjekk ut alle lenker fra semestersiden
 - Bli kjent med undervisningsplanen
- Installer Python og VS code på egen maskin (se info på emnesiden)
- Eventuelt, finn en terminalstue, gjerne med Linux-maskiner:
 - Logg inn med UiO-brukernavn
- Åpne en editor VS Code og et terminalvindu (Terminal (Mac) eller Powershell (Windows))
- Skriv `python` i terminalvinduet og se hvilken versjon du kjører (øverste linje, sannsynligvis 3.7-3.13, kanskje 2.7). Avslutt Python ved å skrive `quit()` eller tast *Ctrl-D*.

- Tok du ikke testen på forkurset, og ønsker å ta den nå?
 - Gjøres på egen hånd, uten hjelpemidler
 - Helt anonymt
 - Teller ikke på karakteren i kurset
- Lenke legges ut under "Beskjeder" på emnesiden, blir tilgjengelig til onsdag 19.08 kl 16.00

Norsk tekst: https://www.pika-platform.org/enter/pika_nor

Engelsk tekst: https://www.pika-platform.org/enter/pika_en