

Velkommen til IN1010 – Objektorientert programmering Våren 2023

Idag:

1. time: Om IN1010 våren 2023
2. time (+ i morgen og to dobbelttimer neste uke):
Java for deg som kan Python

1

Siri Moe Jensen , Stein Gjessing, ++
Institutt for informatikk
Universitetet i Oslo




IN1010 – Objektorientert programmering

I IN1010 skal du lære:

- Å løse et (middels stort) problem ved å lage et program
- Å lage gode (middels store) programmer
- Vite hva et godt program er
- Kunne bruke en del gode programmeringsteknikker
- Objektorientert programmering - Java

Læreboka: Ikke kjøp, finnes i Leganto

- Horstmann: Big Java, Late objects 2013
- Kap 1 – 8 Introduksjon til Java (IN1000 - men Java)
- Kap 9 – arv
- Kap 10 og 11 – GUI (Graphical User Interface)
- Kap 12 – design
- Kap 13 – rekursjon (ikke alt)
- Kap 14 – sortering og søking (bare som eksempelstoff)
- Kap 15 og 16 – datastrukturer (ikke alt)
- Kap 17 – ikke pensum (IN2010)
- Kap 18 – klasser med parametre
- Kap 20 – tråder



Er ikke denne boken
for gammel ?

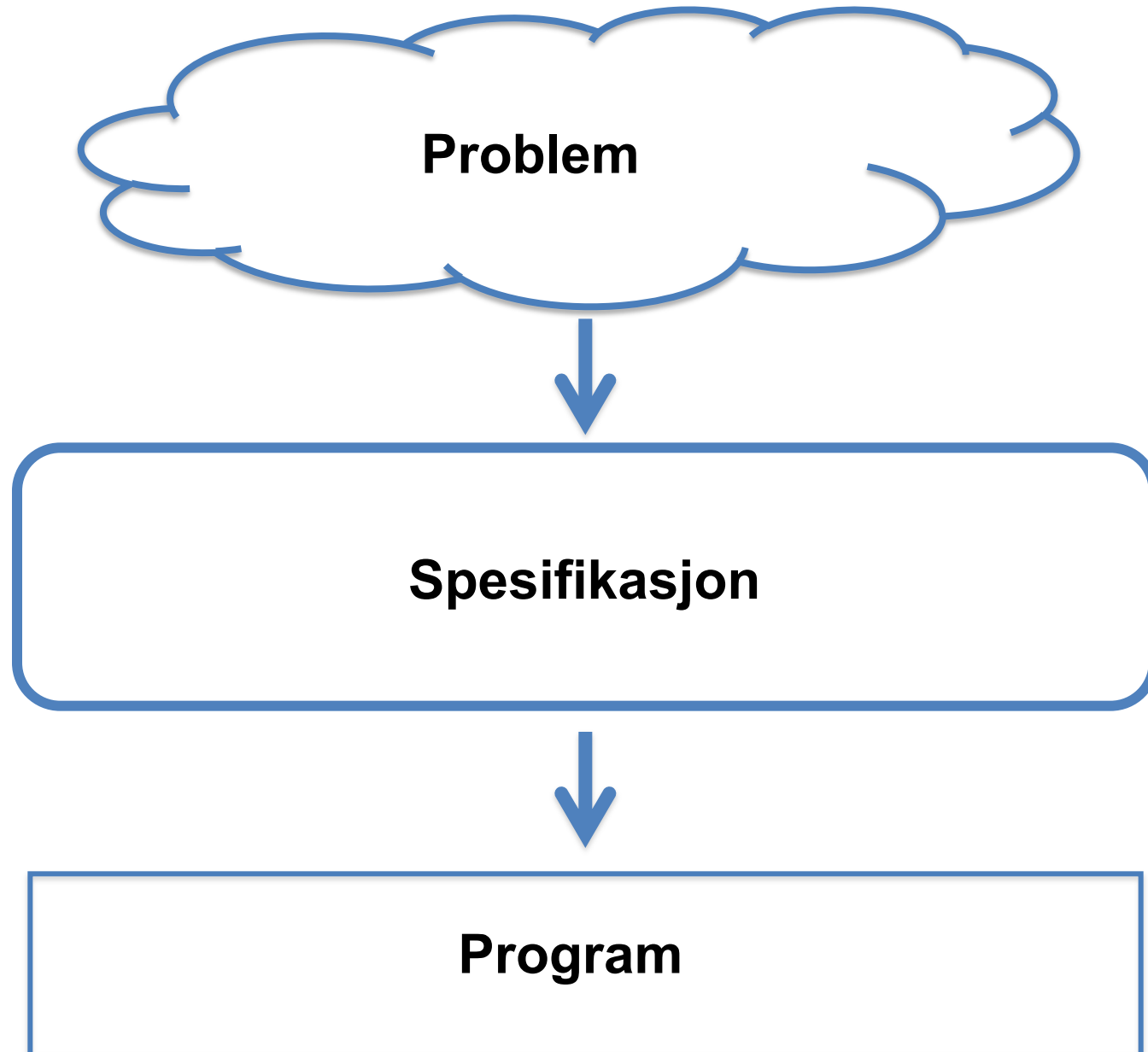
NEI



Men i en annen rekkefølge

Objektorientert programmering

- Ble funnet på for 60 år siden av Ole-Johan Dahl og Kristen Nygaard her i Oslo
 - Språket de fant opp heter Simula-67
 - C++, Smalltalk, Java, C#, . . . bygger på Simula
- Foreleserene i IN1010 (Siri, Dag, Stein) hadde Dahl og Nygaard som lærere
- Pensum er ikke først og fremst læreboka (Horstmann), men det som foreleses (lysarkene). Disse inneholder mye av det vi har lært av Dahl og Nygaard
- Ofte en annen rekkefølge / vinkling enn Horstmann





Eksemplene i IN1010

- Mange av de konseptene vi behandler er viktigst når programmene blir større
- Mange programmer i IN1010 er små
- Konseptene blir introdusert med små, enkle programmer
- Tenk på dette når dere lærer noe nytt
(vi forelesere skal også minne dere på det)
- I noen obligatoriske oppgaver (og noen ganger i plenumstimene) prøver vi å løse litt større problemer



I IN1010 skal du lære

- å programmere: dvs. å *planlegge* hvordan oppgaven som programmet ditt skal løse, skal løses ved hjelp av **datastrukturer (objekter)** og **algoritmer** inne i datamaskinen
- å programmere: dvs. *skrive koden* som lager de **datastrukturene** og implementerer de **algoritmene** du har planlagt
- Derfor skal vi **visualisere** eller tegne **datastrukturer**
 - for å lære algoritmer og datastrukturer
 - for å planlegge programmer
 - for å implementere programmer
 - for å kommunisere om programmet med andre

IN1010 forkunnskapskrav

Du kan løse små problemer i Python vha.

- Variabler og konstanter
 - Verdier, typer / uttrykk
- Tilordninger
- Kontrollstrukturer (valg og løkker)
- Lister/ ordbøker
- Metoder/funksjoner (med parametre og returverdier)
- Klasser
- Objekter
 - Grensesnitt (public – non-public)
 - Referanser



Hvorfor Objektorientert programmering?

- Er den dominerende måten å programmere på
- Alle programmerere må kunne OO
- Kan brukes både til design og implementasjon
- Viser abstraksjon og innkapsling på en god måte
 - Godt pedagogisk verktøy

- Det finnes alternativer til OO-programmering, i alle fall i noen tilfeller
- Vi skal komme tilbake til noen av disse tilfellene senere i IN1010

Denne uken (og neste)

I dag første time: IN1010; "IN1010-kontrakten"

Undervisningstilbud og egne aktiviteter

Andre time i dag, i morgen og neste uke: **Java for deg som kan Python.**

Eget arbeid:

- Les foilene
- Les Horstmann kap 1 – 8.
- Les de to notatene om objekter og klasser i Java av Stein Gjessing
- Gjør de to første ukenes Trix-oppgaver – bli "flytende" i Java
- Løs og lever inn »Innleveringsoppgave 0« (frist 30.1 – frivillig men sterkt anbefalt))
- Delta på gruppetimer – og repetisjonsgruppen ved behov for ekstra gjennomgang
- Delta på lab om du vil jobbe med oppgaver sammen med andre
- Start første obligatoriske oppgave (frist mandag 13. februar) – 2 ukers arbeid



Obligatoriske oppgaver

- Du må ha godkjent alle de 7 obligatoriske oppgavene for å ta eksamen
- Har du godkjent **alle** de obligatoriske oppgavene i IN1010 fra før (dvs. du kunne gått opp til eksamen (inntil 5 år)) trenger du ikke gjøre noen oppgaver på nytt i år
 - Hvis OK fra før: Automatisk inne i systemet
 - Hvis ikke må du gjøre alle oppgavene i år
 - OK å bygge på eget tidligere arbeid, men du må legge inn referanser til eget arbeid
 - Alle oppgavene er forandret i forhold til 2022
 - Du må gjøre ÅRETS oppgaver.
- Hver oppgave er ca to ukes arbeid (ved siden av annet arbeid i IN1010)



Obligatoriske oppgaver

- Etter at et (eller flere) tema er forelest vil det (nesten alltid) bli gitt en obligatorisk oppgave med dette stoffet.
- Du får vanligvis to uker på deg til å løse oppgaven **etter** at stoffet er gjennomgått
 - Noen ganger vil litt av stoffet som trengs i obligen foreleses én uke før fristen
 - Dette vil eventuelt bli markert tydelig i obligen
- Men NB!: Dette betyr at du må jobbe med den obligatoriske oppgaven i det gamle stoffet SAMTIDIG med at du lærer annet nytt stoff.

Obligatoriske oppgaver

Innlevering ca. annenhver mandag*

Stort sett uavhengige oppgaver, men

- Oppgave 3 og 4 bygger på hverandre og på 2
 - Planen er at oppgave 4 skal gjøres i grupper

Å bruke 4 timer per uke på obligatoriske oppgaver er normert ut fra en gjennomsnittsstudent som følger undervisningen og som får til en passe bra innlevering.

* 1. mai er en mandag, så da kommer vi til å flytte litt på innleveringsfristen



Obligatoriske oppgaver

- Skal løses individuelt
 - Diskuter gjerne ideer med andre, men **du skal skrive og forstå all koden selv**
 - Ikke lov å kopiere fra andre
- Du må lese regelverket for obligatoriske oppgaver (lenke fra semestersiden)
- Strengeste reaksjonsform ved fusk: Utestengning ett år
- **Frist for innlevering:** Ca. annenhver mandag kveld kl. 23:59
- Retting: Samretting eller tradisjonell retting
 - informasjon kommer på kurssiden (under obligatoriske oppgaver)



Eksamen

- Onsdag 7. juni 2023 kl 9 – 13 i Silurveien
- Tillatte hjelpemidler: Alle skriftlige, ingen digitale
 - Det betyr at du ikke har tilgang til den elektroniske versjonen av læreboka

Arbeidet i IN1010 - ukehjulet

Undervisningstilbud ≥ 6 timer



Selvstendig arbeid - programmering, 7-8 timer

Teori, Trix, Obliger, Lab

Ressurser: Se [semestersiden](#)

Bøker, notater, lysark, opptak, kollokvier, nettforum, gruppelærere, ..

* En eller to uker kommer til å starte på mandag – følg med på semestersiden



“IN1010 – kontrakten”

Nytt stoff presenteres på forelesningen tirsdag

- Lysark legges ut på forhånd eller rett etter forelesningen
- Foreleserene legger ut opptak så fort som mulig
- Nytt stoff presenteres grundig, først med små enkle eksempler
- Du MÅ (vanligvis) ikke forberede deg på nytt stoff

MEN:

- DET ER DITT ANSVAR AT DU KAN FORRIGE UKES STOFF !!
 - Jobb med øvingsoppgaver i Trix
 - Bruk lysark, lærebok, opptak
 - Gå på grupper, lab - repetisjonsgruppe ved behov (begrenset tid til individuell undervisning på lab av stoff du ikke kan)

Nettforum (Discourse)

- Spørsmål og svar om alt du lurer på om og i IN1010
 - Du finner lenke på kurssiden (og i heading i denne pdf'en)
 - Enkel og grei brukerveiledning ligger på semestersiden
- ⇒ Alle må registrere seg selv ved å gå inn på lenken og logge inn via Feide!!
- Bruk felleskanalene for spørsmål som ikke er gruppe-spesifikke
 - Ikke legg ut obligkode
 - Alle må bidra med hjelp og diskusjon (du *kan* være anonym)


Oppstart av seminartimer og lab

- Seminartimer denne uken (første seminartime onsdag 25. januar):
 - Om du likevel ikke kan komme på din gruppe:
 - Gi beskjed (epost) til gruppelærer om du ønsker å beholde plass (finn gruppelærer under Grupper)
- Repetisjonsgruppe før forelesning (tirsdager kl 12:15 – 14:00)
 - Første gang tirsdag 31. januar.
- Labtimer – mer info kommer på nettsiden
- Fredag ettermiddag: **Fredags-Java** (første gang 27. januar)
 - En lab med fredagsstemning og kanskje superenkel servering?
 - Har noen lyst til å ta med kake?



- Følg med på semestersiden

ifi.uio.no -> **IN1010**



All info på
IN1010s
hjemmeside

- Neste time: Første intro til Java for deg som kan Python
- I morgen: Java med vekt på objektorientering