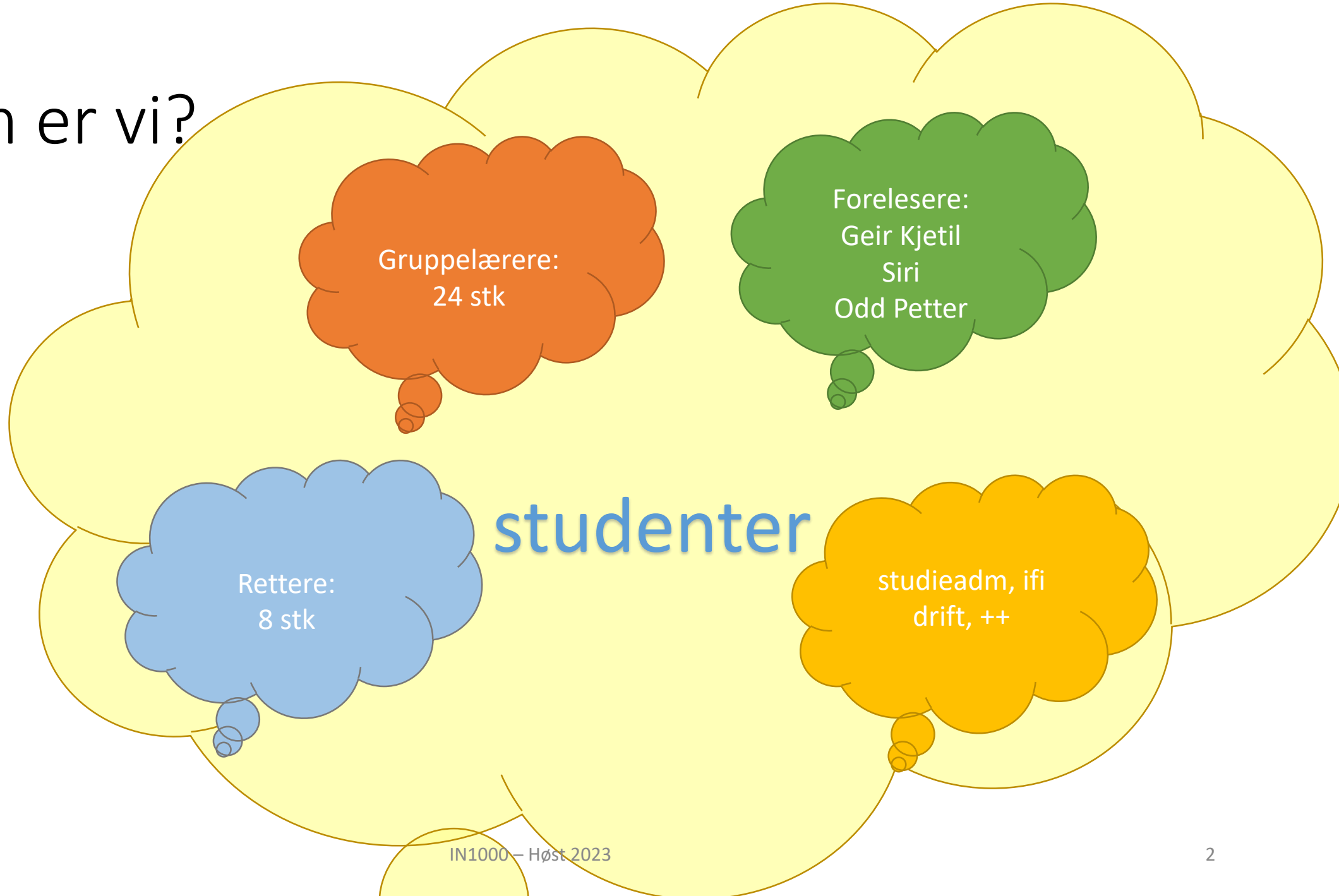
IN1000: Introduksjon til objektorientert programmering

Uke 1

Introduksjon til faget og emnet

Hvem er vi?


Gruppelærere:
24 stk
Forelesere:
Geir Kjetil
Siri
Odd Petter
Rettere:
8 stk
studenter
studieadm, ifi
drift, ++

Hvor enig er du i følgende?



Resultater fra IN1000-forelesning 22.8.23

I dag

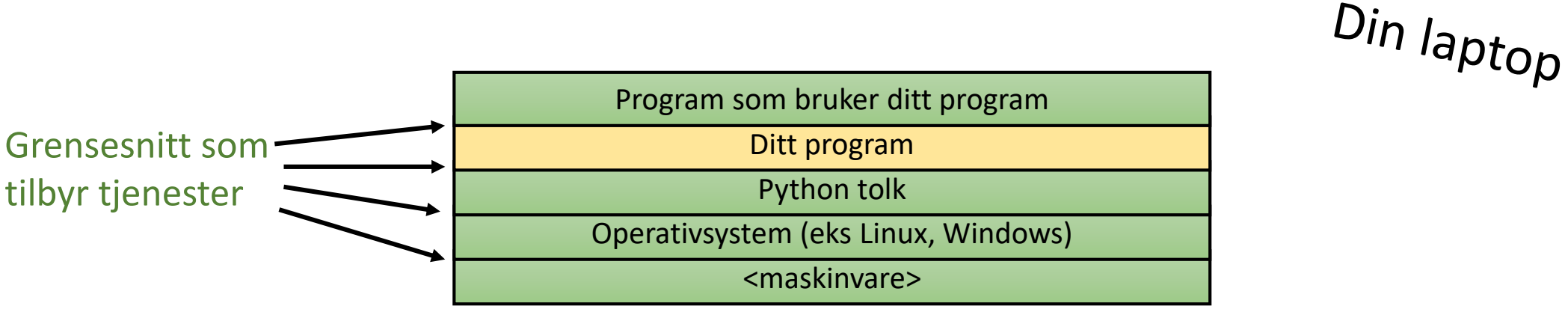
- Informatikk-faget og noen nyttige begreper
- Praktisk informasjon og studietips for IN1000
- <Pause>
- Første innføring i programmering

Informatikk

Informatikk er læren om hvordan datasystemer konstrueres og brukes.

Et datasystem består av en eller flere datamaskiner som kjører programvare og kan være knyttet til et eller flere nettverk for overføring av data

Datamaskin: Lagdeling og grensesnitt



- Hvert lag har sine arbeidsoppgaver, og tilbyr disse som tjenester gjennom et grensesnitt til laget over

- Grensesnittet gjør det mulig å abstrahere bort (ikke kjenne til) *hvordan* laget "under" løser oppgavene sine

Abstrahere/ abstraksjon

Wikipedia:

- "..tenkemåter der en skiller ut enkelte elementer, for å betrakte resten.."
- "..Det er en måte å komme fram til fellestrekk eller felles egenskaper og klassifiseringer ..."

Store Norske Leksikon:

- "å sammenfatte enkeltfenomener under ett begrep idet man ser bort fra avvikende detaljer"

Modell

SNL:

<...> brukes ordet om en skjematisering eller en matematisk modell

Slik brukt innebærer en modell alltid en forenkling basert på et utvalg av de viktigste trekk.

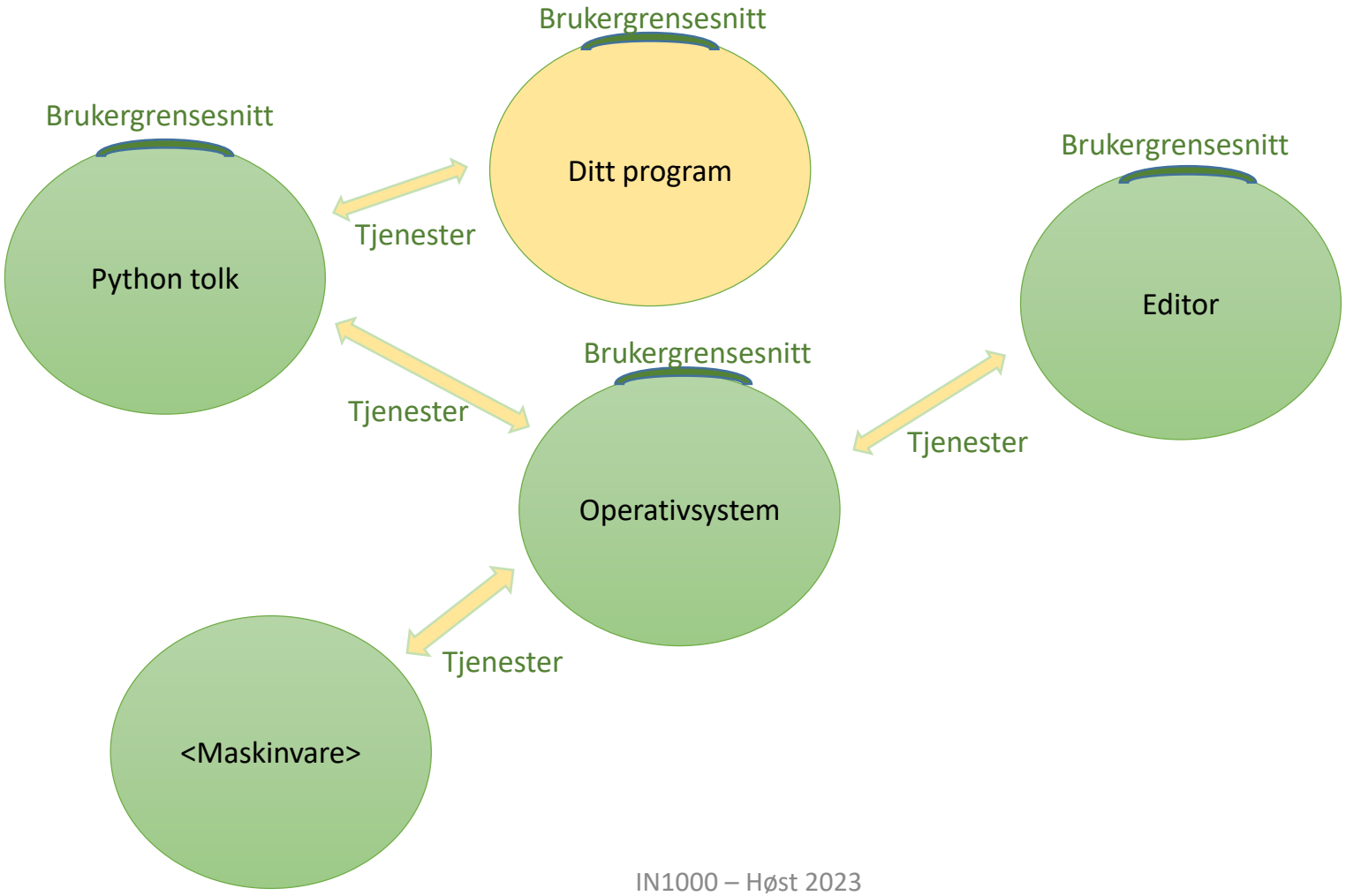
Wikipedia:

Teoretiske modeller gir et sterkt forenklet bilde av virkeligheten.

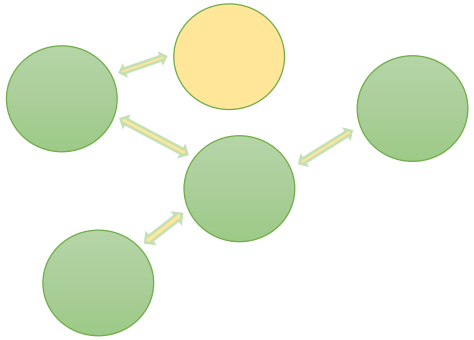
Forenklingen dreier seg om å fokusere på de trekk som anses å være vesentlig for det som skal forklares, samtidig utelukke alt som er uvesentlig



En annen modell av hvordan deler i et system samarbeider



Objektorientert programmering



- Når programmet ditt utføres, foregår det (bak kulissene) vha objekter som samarbeider.
- Et objekt holder rede på data som hører sammen, og kan utføre handlinger knyttet til disse
- *Hvilke* data og handlinger er definert i objektets klassedefinisjon
- Objektene tilbyr tjenester via metoder til andre deler av programmet gjennom sine grensesnitt
- Objektorientering hjelper oss å organisere programmene våre bedre, ved å modellere og simulere komplekse systemer

Vi kommer tilbake til hvordan vi kan opprette og bruke objekter, og hvordan vi kan skrive egne klasser som mønstere for objekter!

Oppsummering

- Informatikk *krever* at vi abstraherer bort forskjeller og kompleksitet vi ikke trenger i det vi jobber med "akkurat nå"
- For eksempel vha modeller og metaforer som beskriver hva som skjer når programmene våre utføres
- Et fenomen kan modelleres på ulike måter avhengig av hvilke egenskaper/ aspekter man er interessert i (trenger å vite noe om)
- Objektorienterte språk har mekanismer som hjelper oss med dette når vi skriver programmer

https://www.uio.no/studier/emner/matnat/ifi/IN1000/

Informatikk-studiet og IN1000

- Mål for emnet
  - Solid grunnlag for videre studier
  - Vekt på grunnleggende, generelle konsepter og begreper
  - Trening i programmering ved hjelp av disse i Python
  - Ikke et hurtigkurs for utviklere – eller i Python

- Ingen krav til forkunnskaper, men helt nødvendig med jevnt arbeid!
  - Gjenspeiles i innleveringsoppgavene

https://www.uio.no/studier/emner/matnat/ifi/IN1000/h23/

Semesteret i IN1000

- Uke 1-6: Grunnleggende elementer i programmering
- Uke 7-11: Objektorientert programmering, større programmer, mer kompleksitet
- Uke 12-14: Prøveeksamen, eksamenstips, repetisjon
- Uke 15: Eksamen (1. desember)

Digital, 4 timer,
kun skriftlige/ trykte hjelpemidler

Innleveringer

- Innlevering 1 og 7 må leveres innen fristen og godkjennes for eksamensrett!

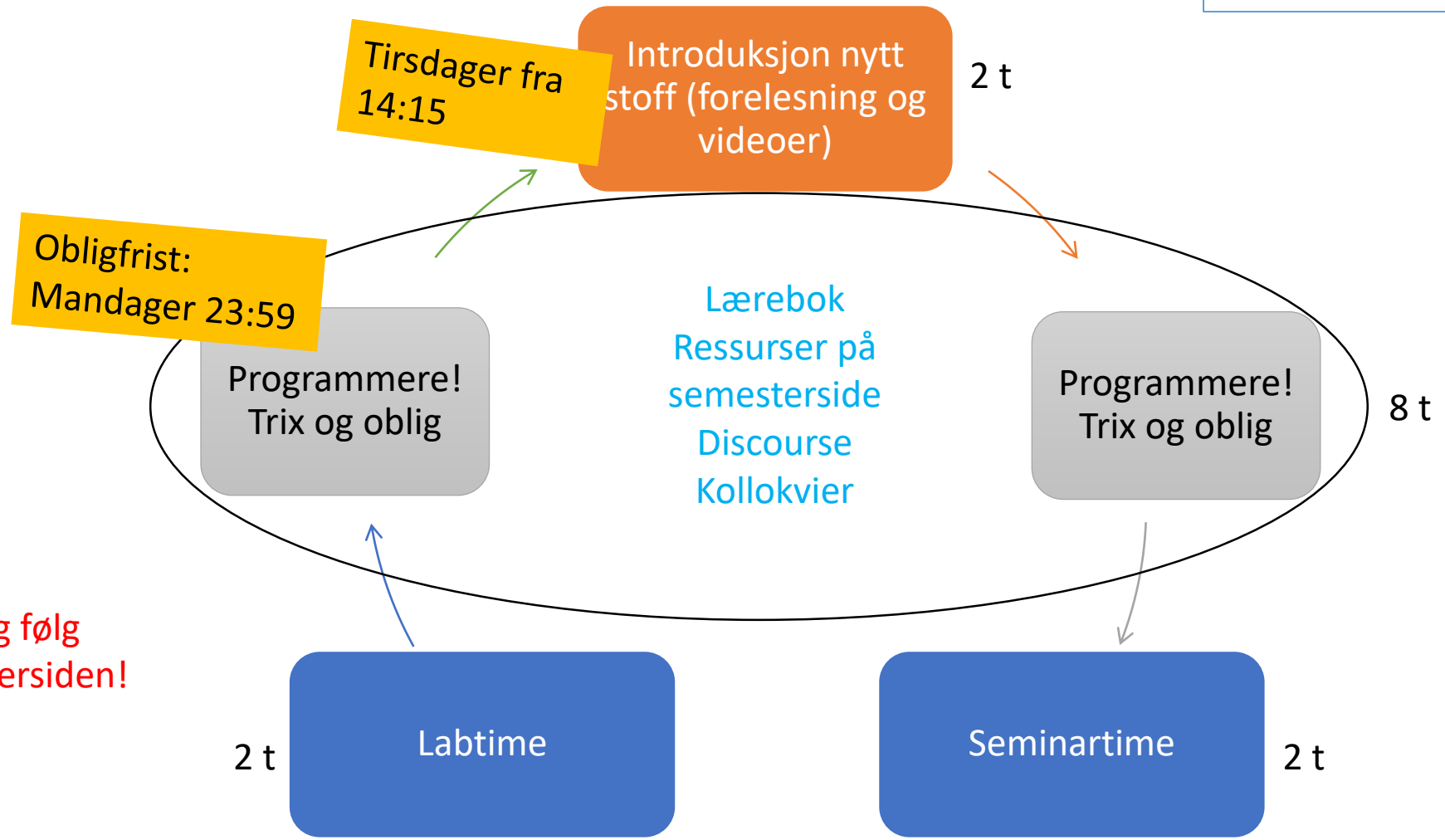
- De andre kreves ikke for å gå opp til eksamen – men er like nødvendige for å **bestå**!
  - Gir deg et inntrykk av hva du har lært av pensum..
  - Nyttige tilbakemeldinger fra retter..
  - ..**hvis** du unngår andres løsninger (og hjelp, før du har jobbet skikkelig med stoffet)

- Alt leveres i Devilry.ifi.uio.no (demo i gruppetime)
  - Harde frister, stenges ved fristen (mandager kl 23:59)
  - Lever gjerne flere ganger, siste gjelder (NB: Filnavn i oppgaveteksten)

Alt du trenger å vite finner du under I**nnleveringer** på semestersiden

Arbeidet i IN1000 – ca 13 t/ uke

Dessuten:
- Fredags-Python: Fredag 14:15-16 lab, av og til med kaffe ++
- Flere tilbud kommer – følg med på semestersiden

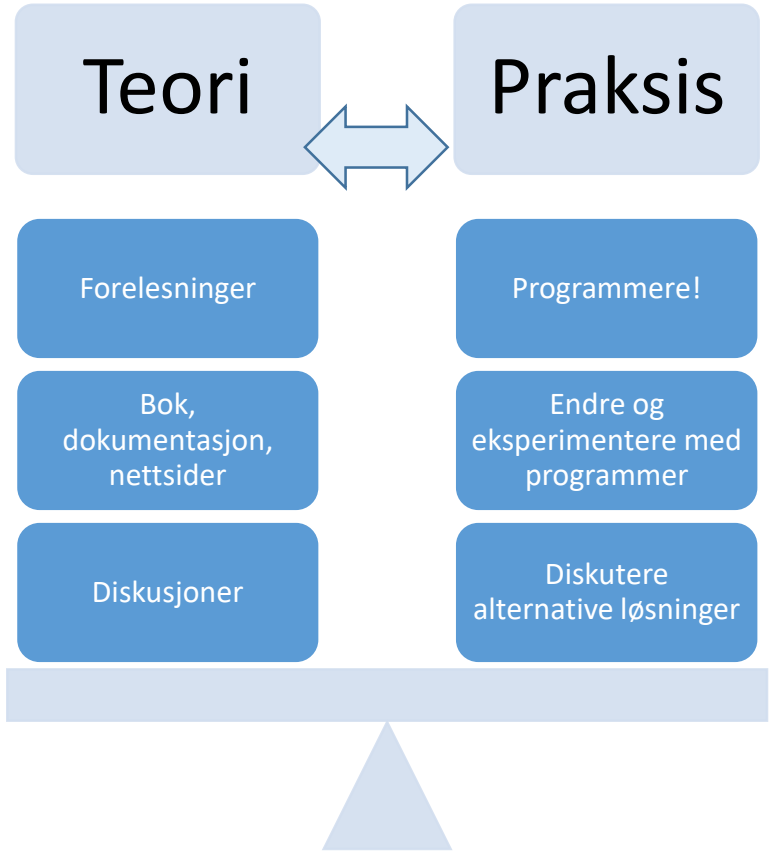


Bli kjent med og følg med på semestersiden!

Programmering er et kreativt håndverk

- Teori i IN1000 handler om å lære
  - nye konstruksjoner og nyttige fremgangsmåter
  - prinsipper som øker kvaliteten på programmene du skriver
  - et begrepsapparat som gjør det raskere å lære nye språk og anvendelser

- Du lærer ikke å programmere uten å skrive (mange) programmer
- Du blir ikke en effektiv programmerer som kan kommunisere med andre og raskt gjenbruke kunnskap i nye situasjoner *uten å bruke tid på teori*



Tips for programmeringsstudier

- Programmering handler om å definere og løse problemer/ finne fremgangsmåter, dvs
  - Læringen skjer mens du klør deg i hodet
  - Lær å like det ☺
  - Ikke gi opp – selv om det av og til **tar tid**!



- Start koding selv om du ikke forstår alt, eksperimenter
- Men finn alltid ut hvorfor en feil "forsvant"!

Praksis

Teori

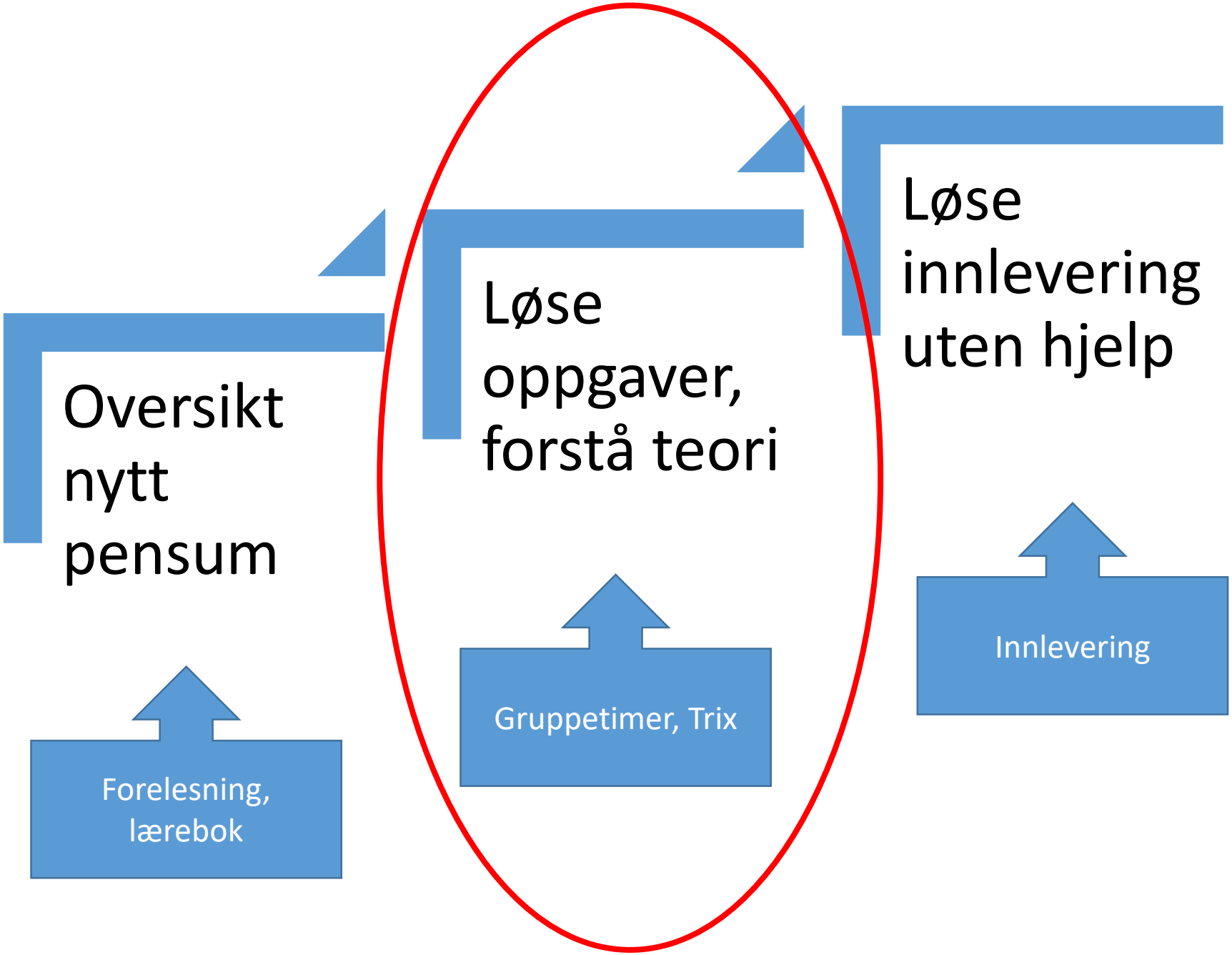


- Jobb sammen med andre, diskuter løsninger og "hva skjer her"
- Jobb alene, svett over problemer ☺

Gruppetimer, Trix

Innlevering!

- Du kan ikke hoppe over pensum i programmering. Husk jevnt arbeid!


Oversikt nytt pensum
Løse oppgaver, forstå teori
Løse innlevering uten hjelp
Hvor mye klarer jeg på egen hånd?
Forelesning, lærebok
Gruppetimer, Trix
Innlevering

Forkunnskaper, lyst på mer utfordringer?

- Noen fallgruver
  - Følg med på pensum og løs innleveringene så du oppdager hull!
- Tilbud med alternative (mer krevende) oppgaver
  - Starter opp uke 3, følg med på IN1000 semestersiden
- Overskudd til ekstra studiepoeng?
  - IN1150 – Logiske metoder for informatikk (nettkurs i høstsemesteret)

Nasjonal forkunnskapsprøve i programmering

- Mange universiteter og høyskoler
- Anonym!
- Vil gjerne at *alle* prøver seg, selv om du ikke har noen forkunnskaper
- Kan bruke opp til 2 timer. Antakelig trengs under 1 time
- Viktig: INGEN hjelpemidler – ikke hjelp, internett, bøker eller kjøre kode
- Skal ikke deles med andre
- e-post med lenke kommer når forelesningen er slutt – åpen ut morgendagen