



Velkommen til INF 1000 – høsten 2009

Grunnkurs i Objektorientert Programmering
Institutt for Informatikk
Universitetet i Oslo

Arne Maus, Ole Christian Lingjærde og Arild Waaler

1



Plan for denne uka:

- I dag:
 - Intro til INF1000
 - Praktiske forhold bla. bruk av Joly
 - Registrering av oppmøte i pausen (miste plass ?)
 - De første program + variabler, deklarasjon og tilordning
- Gruppeundervisningen (grupper 1-10) starter:
 - Neste uke fom. mandag 24. aug.
- Mål: Du greier å bruke maskinen i løpet av denne eller neste uke og har forsøkt å lage et program

2



Mål for INF1000:

- Gi grunnleggende forståelse av noen sentrale begreper, problemstillinger og metoder innen informatikk
- **Lære å programmere**
- Gi noe innsikt i datamaskiners muligheter og begrensninger
- Lære noe om samfunnsmessige konsekvenser av bruk av informasjonsteknologi

3



Hva er INF1000

- Felles innføringskurs i Objektorientert programmering for ca 8 bachelor-programmer (MatNat, HF, Jus) , 5 om høsten, 4 om våren
- 10 studiepoeng
- ca. 420 studenter (høst) og 120 (vår)
- Et frittstående introduksjonskurs for deg som vil lære å programmere Java og kanskje ta flere kurs senere.
- Programmering videreføres i INF1010 i vårsemesteret 2010
- To begynnerkurs INF1000 og INF1100, likheter og forskjeller:
 - INF1000 – lærer å analysere problemer og lage løsninger på egen hånd, i form av programmer i Java
 - INF1100 – lærer å beregne og visualisere gitte matematiske uttrykk med datamaskin i Python og Matlab

Uansett: Mange målgrupper i INF1000– felles kurs, nesten felles oppgaver og eksamen (oblig 3 i to varianter)

4



INF1000

- Innhold:
 - Litt datateknologi
 - Noe tekstbehandling
 - Mye programmering
- Verktøy:
 - Datamaskiner med Unix og Windows på Blindern eller hjemme-PC med Windows
 - Tekstbehandlingssystemet Emacs på Unix og f.eks Emacs eller TextPad på PC
 - Programmeringsspråket Java
- N.B: Å lære å programmere er mer enn det å lære et programmeringsspråk

5



Dine forutsetninger

- Data-kurs ?
- Har lært det selv ?
- Matematikk ?
- Noe annet ?

6



Advarsel:

- Mye å gjøre
- Mye ferdighetstrening
- Mange oppgaver:
 - Nytt sett øvelsesoppgaver hver uke
 - 4 obligatoriske oppgaver, hvorav **alle** må løses og godkjennes for å kunne gå opp til eksamen – første 'oblig' skal være levert 4. sept. – om 2 ½ uke

7



Undervisningen høsten2009

- Forelesninger:
 - 2 timer 'hver' uke og tirsdag 14.15- 16.00, v/ Ole Christian Lingjærde, Arne Maus , Arild Waaler
- Inndelt i ca. 10 grupper
 - Oppgavegjennomgang i 2 t/u
 - Gjennomgang av ukeoppgaver
 - Første gang neste uke
 - Terminaltimer , 2 timer terminal/hjelp i uka .
 - Hjelp til å løse oppgaver praktisk på terminal – ukeoppgaver og oblig'er.
 - Første hjelpelærerruke neste uke
- Selvstudium :
 - Lesing, programmering – også løse egne oppgaver, mange timer pr. uke
- Selvtest på nettet (flervalgstest)
 - test om du har forstått siste forelesning (anonymt)

8

Undervisningsmaterieell, del I

- Lærebok - kjøpes i Akademika:
 - Brunland, Hegna, Lingjærde og Maus: **Rett på Java**, 2. utg (Universitetsforlaget, 2007)
- Følgende lastes snart ned via hjemmesida til kurset : <http://www.ifi.uio.no/INF1000/h09>
 - Unix for nybegynnere
 - kompendium av Dag Langmyhr
 - Local guide til Emacs
 - kompendium av Dag Langmyhr
 - Informasjonsteknologi, vitenskap og samfunnsmessige virkninger
 - kompendium av Arne Maus

9

Undervisningsmaterieell, del II

- Hjemme DVD for PC med mye nyttig programvare
 - Bla. Emacs, TextPad, Java, Pyton og en rekke andre programmeringsspråk
 - Deles ut gratis om noen uker
- Kopier av lysarkene fra forelesningene
 - Dere skrive dem ut selv via klikking på hjemmesida (betaling!)
 - Kan lastes ned over nettet fra hjemmesida og leses på maskin eller skrives ut.
 - Dere betaler litt for utskrifter ut over de første 100 ark

10

Oblig'er og andre oppgaver

- En obligatorisk (programmerings-) øvelse ca. hver tredje uke
 - **Individuell** besvarelse !
 - Leveres hjelpelærer til retting/godkjenning før fristen vi **Joly**.
 - Hjelp og tips fra medstudenter tillatt, men kopi strengt forbudt (like besvarelser – vil bli sjekket av et program og bli behandlet som fusk)
 - **25** studenter ble meldt for fusk/kopiering h2007 og v2008 (tatt av Joly)
 - Alle som ble tatt v2008 ble utvist (minst et ½ år)
 - Kan hende du må forbedre ditt løsningsforslag
 - Dere som har tatt inf1000 før: Gamle godkjenninger fortsatt gyldige. Sjekk med gruppelærer
 - to varianter av oblig 3 (administrativ eller matematisk) ellers samme obliger for alle.
 - **Alle obligene skal de leveres via Joly-systemet**
- + øvelsesoppgaver, nytt sett hver uke
- Flere enn du greier å gjøre
- Gjennomgås på gruppene
- Løsningsforslag gis
- Gå på gruppene - det er der man får kontakt med andre studenter - og man lærer mye av hverandre (og gruppelæreren)

11

<http://obelix.ifi.uio.no:8080/>

Joly: Velg kurs

Velkommen til Joly - online innlevering

Denne assistenten vil ta deg gjennom de nødvendige stegene for å levere en oblig. Først må du velge hvilket kurs du skal levere oblig i:

Kurs: INF1000

videre >

Hva hvis ??

- Jeg skal sende inn forbedret oblig
 - OK – alle blir lagret (og gruppelærer retter selvsagt den siste)
- Hvordan legger jeg inn gruppenummer
 - Systemet finner selv hvilken gruppe du er påmeldt (og skal sende obliger til)
- Joly-systemet er nede når jeg skal levere
 - Levér på vanlig e-post til din gruppelærer (så legger hun/han) inn obligen din
- Hvis jeg ikke er registrert som student på gruppa
 - Da kan du fortsatt levere, men besvarelsen blir ikke lagret i basen
 - se neste foil
- Kan jeg levere hjemmefra
 - Ja, hvis du har VPN eller 'Remote Desktop' (tit: wind.ifi.uio.no)

13

Eksamen

- 30.nov. kl. 1430-1700
 - Trekkefrist er 14 dager før eksamen.
- *Alle* skriftlige hjelpemidler tillatt
- Karakter: A, B,..., E og Stryk (F)
- Altså for å stå i INF1000 må **både** alle de 4 obligene være godkjent **og** eksamen består
- Karakteren bestemmes av eksamensbesvarelsen.

14

Tilgjengelig utstyr

- Ved Ifi og MatNat:
 - ca. 500 datamaskiner i nett
 - ca. 10 store laserskrivere
 - Tilgjengelig for INF1000 (og andre laveregradskurs) på Ifi:
 - ca. 240 datamaskin-arbeidsplasser (m. køsystem)
 - Nettet (WWW), med tilgang til elektronisk post og Internett hvis logget inn
 - Pålogget hjemmefra, tre muligheter:
 1. 'Remote Desktop' fra hjemmePCen (til windows.ifi.uio.no). Lett, men må settes opp for å kopiere filer til lokal C: disk på hjemmePCen
 2. Innlogging hjemmefra via USIT og da som Ifi-bruker. **Sett opp en VPN-forbinnelse** Se: <http://www.usit.uio.no/it/student/hjemmemaskin.html>
 3. Adgang til hjemmeside og alt annet INF1000 stoff via WWW, men koblet opp mot en annen nettleverandør enn Ifi/USIT.
- Forskjellen:** Små, men som innlogget via USIT har du adgang til å kopiere filene dine på Ifi til/fra hjemmet, samt sende e-post som Ifi-student. Liten grunn til å bli ifi-bruker hvis du aldri er på Blindern. Se <http://www.ifi.uio.no/it/>

15

5 informatikk "terminal-stuer":

- Abel
 - Stor, men også fullest ,
 - VB
 - Enda større.
 - Informatikk-bygget:
 - Mindre og bedre plass
 - + Muligheter for bruk av andre MatNat-maskiner på Bio, Fysikk, Kjemisk
 - Best plass og minst
 - Terminalvaktene på Abel og VB bygget hjelper deg !
 - Både Windows og Linux (Unix) maskiner og tynne klienter
- <http://termvakt.uio.no/wiki/>
 - **VB (betjent) 50 +73 maskiner**
man-tors: 0800-2030 fre: 0800-1830 lør:
1000-1600, søn: 1000-1730 (kort+kode)
 - **Abel (betjent) 68 maskiner**, man-fre:
0800-2030
 - **Fysikk (betjent) 40 – oppussing, snart ferdig**
man-tors: 0800-2000 fre: 0800-1800
 - **IFI (ubetjent) 40 m**, man-fre: 0700-1700
lør: stengt
 - **IFI (ubetjent) (med kort+kode)**
man-fre: 1700-2100 lør: 0800-1400
 - trådløst nett: **uio + vanlig innlogging**

16

Noen vanlige spørsmål:

- Hvordan bytte gruppe?
- Kan jeg etteranmelde meg - JA
- Når/hvor få passord - (i posten !)?
- Hvordan/hvorfor bli fratatt passord ?
- Er det mulig å bruke egen PC ?
- Må jeg ha hjemme-PC ?
- Hvordan få Ifis DVD-plate med bl.a Emacs, TextPad og Java (til hjemme-PC)?
- Hvordan får jeg beskjeder fra Ifi/kurset?

17

Brukernavn og Passord

- For å få adgang til maskinen trenges to opplysninger
 - **Brukernavn** (en kortform av navnet ditt) – dette er offentlig. Jeg (Arne Maus) har f.eks brukernavnet: *arnem*
 - **Passord** (hemmelig) – tastes inn etter at du har oppgitt brukernavn. Gir sikkerhet for deg.
- Hvordan få brukernavn og passord ?
 - I posten når du er semesterregistrert (eller av termvakt hvis du ikke finner ditt, eller hos: *Huston – 2 etg Inf-bygget*)
 - Kontoen virker ikke før du har betalt semesteravgift!
 - Trenger du adgang til Ifis maskiner
 - NEI – hvis du aldri er på Blindern og har eget internett abonnement
 - JA – ellers
- Se: <http://www.usit.uio.no/it/student/>

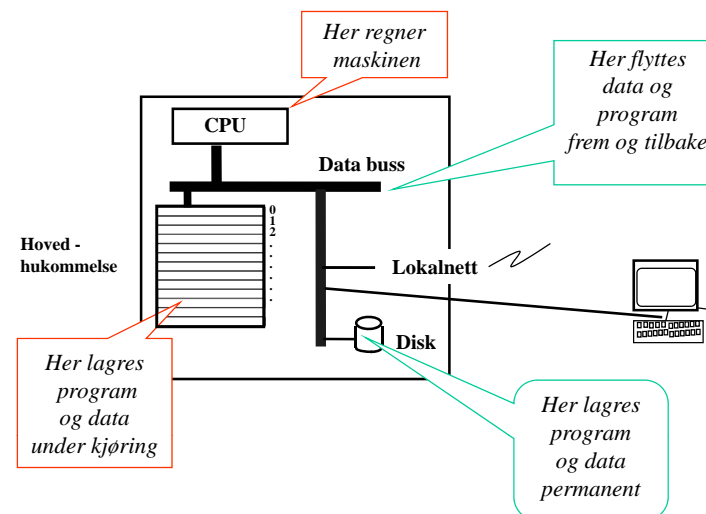
18

Det første program

- Hva er en datamaskin
- Hvordan får vi den til å gjøre det vi vil
- Et program er en oppskrift til maskinen
- Hvordan får vi programmer inn i maskinen
- Hvordan får vi maskinen til å kjøre programmet
- Et første program

19

Hva er en datamaskin





Hvordan får vi maskinen til å gjøre det vi vil

- Vi gir den en rekke ordre
- Maskinen er ganske innskrenket, men kan noen få typer ordre:
 - Les inn et tall (fra tastatur)
 - Skriv ut en tekst (til skjerm, disk,..)
 - Legg sammen to tall
 -
- For å få gjort det vi vil, ber vi maskinen utføre et antall slike ordre/handlinger (én etter én, ovenifra og nedover)
- Denne rekken av ordre kalles et **program**

21



Et program er en oppskrift til maskinen

- Vi kjenner andre typer oppskrifter:
 - matoppskrift
 - strikkeoppskrift
 - pianonoter
 -
- Et program er en oppskrift til en datamaskin
- Med *noter* lager man ulike melodier ved å kombinere et mindre antall muligheter for lyder fra pianoet
- Med *programmering* kan men lage alle mulige programmer ved å kombinere et begrenset sett av mulige operasjoner i datamaskinen
- Husk: Det er enklere å *følge* en oppskrift enn å *lage* en oppskrift selv (som dere skal lære i INF1000)

22



Hvordan får vi våre programmer inn i maskinen

N.B.: Det finnes allerede en rekke programmer inne i datamaskinen:

- operativsystemet
- (program-) editoren (emacs, TextPad, WordPad,...)
- oversetteren (kompilatoren)
- kjøre-programmet
-

Det er disse programmene som *hjelper* deg til å få ditt program inn i maskinen

23

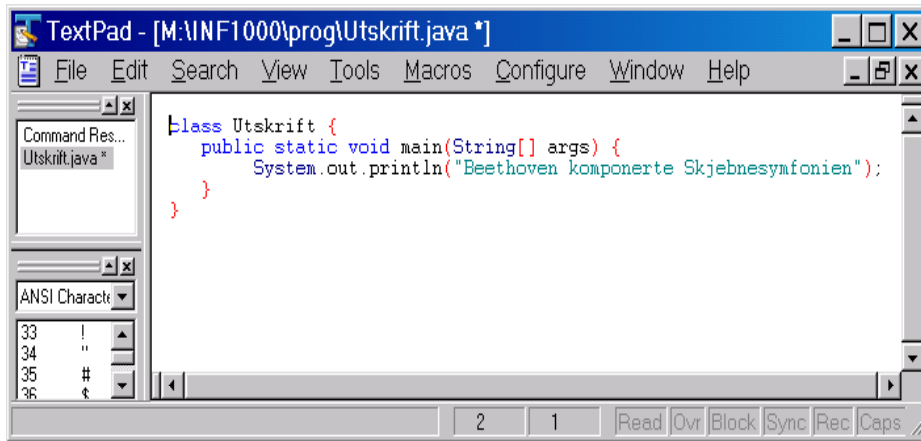


Programmering:

- Vi skriver våre programmer på en måte som er lettest for oss mennesker (til editoren)
- Denne skrivemåten kalles et programmeringsspråk
- En programtekst skrevet i et slikt programmeringsspråk kan:
 - lett oversettes (av oversetteren) til enkle operasjoner,
 - som lagres i hovedhukommelsen og
 - så kjøres (av kjøre-programmet)
- Det er mange programmeringsspråk - det vi bruker i INF1000 heter **Java**

24

Et første program i Java (– her inne i TextPad)

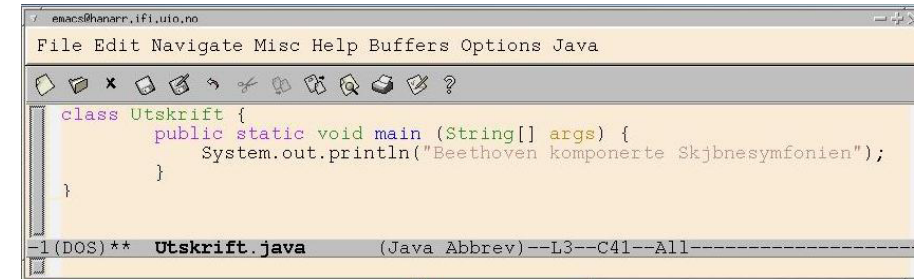


The screenshot shows the TextPad editor window titled "TextPad - [M:\INF1000\prog\Utskrift.java *]". The menu bar includes File, Edit, Search, View, Tools, Macros, Configure, Window, and Help. The code in the editor is:

```
class Utskrift {  
    public static void main(String[] args) {  
        System.out.println("Beethoven komponerte Skjebnesymfonien");  
    }  
}
```

The status bar at the bottom shows "2 1 Read Ovr Block Sync Rec Caps".

Et første program i Java (– her inne i emacs)



The screenshot shows the Emacs editor window titled "emacs@hanarr.ifi.uio.no". The menu bar includes File, Edit, Navigate, Misc, Help, Buffers, Options, and Java. The code in the editor is:

```
class Utskrift {  
    public static void main (String[] args) {  
        System.out.println("Beethoven komponerte Skjebnesymfonien");  
    }  
}
```

The status bar at the bottom shows "1 (DOS) ** Utskrift.java (Java Abbrev)--L3--C41--A11".

26

Én utskriftsordre i dette programmet (+ noe uforstålig)

```
class Utskrift {  
    public static void main(String[] args) {  
        System.out.println("Beethoven komponerte Skjebnesymfonien");  
    }  
}
```

- Et Javaprogram består av minst:
 - En klasse – her: class Utskrift
 - En metode som heter **main** og med Java-ordene: **public static void** foran seg + parameter i parentes (String[] args)
 - dette forklares senere
- Inne i metoden **main** er det én eller flere ordre – her:
`System.out.println("Beethoven komponerte Skjebnesymfonien");`

27

Kompilering (=oversetting) og kjøring (av det oversatte)

>javac Utskrift.java

Her oversettes programmet og oversettelsen lagres i fila: Utskrift.class

>java Utskrift

Her ber vi om at det oversatte programmet (i Utskrift.class) skal kjøres

Beethoven komponerte Skjebnesymfonien

Denne linja er resultatet av kjøring av programmet

28

Programmering generelt

- Vi skriver programmet som en tekst i en editor (TexPad, emacs,...)
- Vi lagrer filen med programmet lik navnet på klassen og med **java** etter punktum – her: **Utskrift.java**
- Vi lar kompilatoren **javac** oversette **.java** filen og legge oversettelsen i en ny fil - her: **Utskrift.class**
- Vi starter opp kjøresystemet **java** med **Utskrift** som parameter på samme linje (den forstår at dette er **Utskrift.class**)
- Kjøresystemet leser så denne og utfører de instruksjonene som ligger på **.class** fila - her: **Utskrift.class**
- Kommandoene som ligger i **main** blir da utført,
 - en etter en
 - ovenfra og nedover (til vi har utført siste ordre i main)

29

Et litt større program – tre linjer utskrift med kompilering og kjøring

```
class Utskrift2 {  
    public static void main(String[] args) {  
        System.out.println("Arne har aldri komponert en symfoni");  
        System.out.println("Beethoven komponerte Skjebnesymfonien");  
        System.out.println(" -----*****-----");  
    }  
}
```

Kompilering og kjøring:

```
>javac Utskrift2.java  
  
>java Utskrift2  
Arne har aldri komponert en symfoni  
Beethoven komponerte Skjebnesymfonien  
-----*****-----
```

30

Oppsummering om programmering

1. Analyser problemet
 1. Hva skal vi få maskinen til å gjøre
2. Skriv Java-programmet som gjør det
 1. Bruk en tekst-editor og lagr det som en **.java** – fil
3. Kompiler og kjør programmet
 1. **javac** og **java**

Hvis du har skrevet programmet litt feil, får du (mange) feilmeldinger fra kompilatoren. Da må du rette opp **.java** fila og gjenta kompileringen til det blir riktig, og så kan du kjøre programmet.

31

