



Velkommen til INF 1000 – høsten 2005

Grunnkurs i Objektorientert Programmering
Institutt for Informatikk
Universitet i Oslo

Arne Maus og Arild Waaler

1



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

2



Hva er INF1000

- Felles innføringskurs i Objektorientert programmering for ca 13 bachelor-programmer ved MatNat – 10 studiepoeng
- ca. 750 studenter
- Et frittstående introduksjonskurs for deg som vil lære å programmere Java og kanskje ta flere kurs senere.
- Videreføres i INF1010 i vårsemesteret 2006
- Tas ofte sammen med INF1040 – digitale media og/eller et matte-kurs (MAT1000 / MAT1100 / MAT-INF1100)

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

3



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 Wordpad /TextPad på PC
 - Programmeringsspråket Java
- **N.B:** Å lære å programmere er mer enn det å lære et programmeringsspråk

4



Dine forutsetninger

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

5



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 9 sept. – om 2 ½ uke

6



Undervisningen høsten 2004

- Forelesninger:
 - 2 timer 'hver' uke (tirsdag 12.15-14.00 midlertidig til uke 41 eller..) og tirsdag 14.15- 16.00) v/ Arne Maus og Arild Waaler
- Øvelser i grupper – ca. 30 grupper :
 - 2 timer teori, 2 timer terminal/hjelp i uka ved ca. 30 (?) hjelpelærere.
 - Første hjelpelærerule neste uke
- Selvstudium :
 - Lesing, programmering – også løse egne oppgaver, mange timer pr. uke
- Selvttest på nettet (flervalgstest)
 - test om du har forstått siste forelesning (anonymt)

7



Undervisningsmaterieell, del I

- Lærebok - kjøpes i Akademika:
 - Brunland, Hegna, Lingjærde og Maus: *Rett på Java, 2. utg* (Universitetsforlaget, 2005)
- Følgende lastes snart ned via hjemmesida til kurset :
<http://www.uio.no/studier/emner/matnat/ifi/INF1000/h05/>
 - Unix for nybegynnere
 - kompendium av Dag Langmyhr
 - Local guide til Emacs
 - kompendium av Dag Langmyhr
 - Informasjonsteknologi, vitenskap og samfunnsmessige virkninger
 - kompendium av Arne Maus

8

Undervisningsmaterieell, del II

- Hjemme CD for PC med mye nyttig programvare
 - Bla. Emacs, Java, og en rekke andre programmeringsspråk
 - Deles ut gratis om ca. 2 uker
- Kopier av lysarkene fra forelesningene
 - Deles ut gratis på forelesningene i august – etter det må 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

9

Oblig'er og andre oppgaver

- En obligatorisk (programmerings-) øvelse ca. hver tredje uke
 - **Individuell** besvarelse !
 - Leveres hjelpelærer til retting/godkjenning før fristen (skal bruke e-post)
 - Hjelp og tips fra medstudenter tillatt, men kopi strengt forbudt (like besvarelser – vil bli sjekket av et program og bli behandlet som fusk)
 - 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.
- + 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)

10

Eksamen

- 2. desember kl. 1430-1730
 - 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.

11

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)
 - Nett (WWW), med tilgang til elektronisk post og Internett hvis logget inn
 - Pålogget hjemmefra, to muligheter:
 1. Innlogging hjemmefra via USIT og da som Ifi-bruker. Sett opp en VPN-forbinnelse Se: <http://www.usit.uio.no/it/student/hjemmemaskin.html>
 2. 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/>

12

7 "terminal-stuer":

- Abel og VB,
 - Størst, men også fullest ,
- PO-bygget og Informatikk-bygget:
 - Mindre og bedre plass
 - Informatikk er åpen hele døgnet !
- + Muligheter for bruk av andre MatNat-maskiner på Bio, Fysikk, Kjemisk
 - Best plass og minst
- Terminalvaktene på Abel, Bio og VB-bygget hjelper deg !
- Både Windows og Linux (Unix) maskiner

13

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 CD-plate med bl.a Emacs og Java (til hjemme-PC)?
- Hvordan får jeg beskjeder fra Ifi/kurset?

14

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) – testes 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)
 - 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/>

15

Student-IT ved UiO - Microsoft Internet Explorer

Address <http://www.usit.uio.no/it/student/>

UNIVERSITETET I OSLO

Om UiO Studier Studentliv Forskning For ansatte Oppslagstavla Bibliotek UiO-nettsider Søk

IT-TJENESTER

Student-IT ved UiO

[In English](#)

Velkommen til IT-tjenestene for studenter ved Universitetet i Oslo.

Å studere ved Universitetet i Oslo innebærer utstrakt bruk av IT. Som student vil du være avhengig av tilgang til IT-tjenester og IT-verktøy for å gjennomføre utdanningen. Du får tildelt et brukernavn med et tilhørende passord. Dette gir deg mulighet til å bruke en rekke IT-tjenester og IT-verktøy i studiearbeidet. Disse sidene gir deg en oversikt over de mulighetene du har, inkludert en oversikt over hvilke regler og bestemmelser som gjelder for bruken av IT-tjenestene.

Aktuelt:

Kopiavgift og betaling for utskrifter via UIOs sentrale utskriftssystem

Fra og med høstsemesteret 2004 skal studenter ved UiO betale for både kopier som deles ut på forelesninger og kurs, samt for hver enkelt utskrift de foretar. Prisen per utskriftstrykk er satt til 30 øre, (s/h A4-utskrift, egne priser gjelder for fargeutskrift og andre spesialutskrifter).

Kopiavgiften er satt til 100 kroner for heltidsstudenter, 50 kroner for deltids- og emnestudenter. Alle aktive studenter må betale denne avgiften, for studenter som unnlater å gjøre dette faller retten til å skrive ut via UIOs utskriftssystemer bort.

Les mer under [Utskrift](#)

Nye studenter får tilsendt et brev med brukernavn og passord. Ta godt vare på brevet til du har logget deg inn på UIOs datanett første gang og byttet passord.

Brukernavn og passord
Hva er brukernavn – Hva er passord – Bytte av passord

Retningslinjer og bestemmelser
IT-reglementet – Lisensregler for programvare – Regler for studentbynett – Lokale retningslinjer – Skikk og bruk med e-post – Regler for personlige hjemmesider – Oppfølging

Plan for denne uka:

- I dag:
 - Intro til INF1000
 - Praktiske forhold
 - Egen intro. ved Morten Dæhlen, instituttbestyrer Ifi
 - Registrering av oppmøte (miste plass ?)
 - Et første program
 - Om første bruk av datamaskin (Unix, Windows og emacs)
- Gruppeundervisningen starter:
 - Neste uke fom. mandag 29 august.
- Mål: Du greier å bruke maskinen i løpet av denne eller neste uke og har forsøkt å lage et program

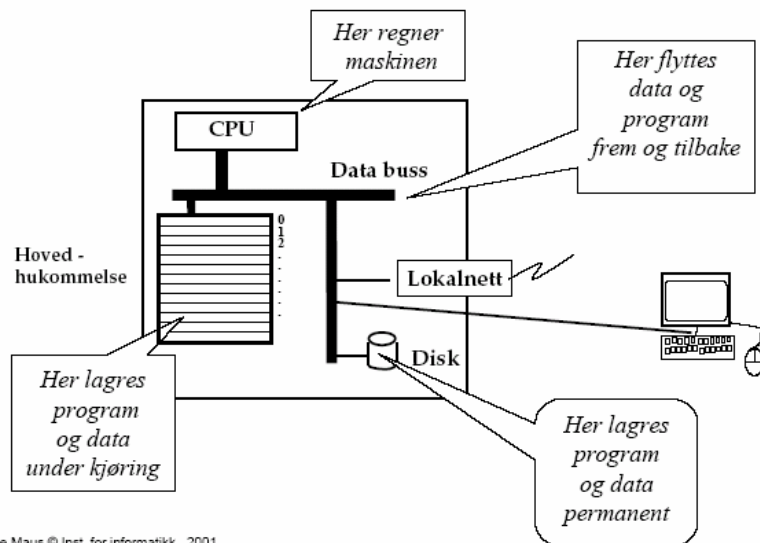
17

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

18

Hva er en datamaskin



Arne Maus © Inst. for informatikk, 2001

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)
- Denne rekken av ordre kalles et **program**

20

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)

21

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

22

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**

23

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

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

En utskriftsordre i dette programmet (+ noe uforståelig)

```
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 metode `main` er det én eller flere ordre – her:
`System.out.println("Beethoven komponerte Skjebnesymfonien");`

25

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

26

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`)

27

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
-----*****-----
```

28

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.
Mer om feil i neste forelesning.

29

Unix og Windows

- To familier av operativsystemer
 - Holder orden på filer, kjøring av programmer, kommunikasjon (til: skjerm, tastatur, mus, nett, skriver,...mm)
- Windows fra ca. 1985, mest vinduer med *pek og klikk*, men også mulig med kommandoer, finnes i to varianter
 - Foreldet: Windows 95/98/98se/Me
 - Windows NT 3.41/NT 4.0/ Windows 2000 /Windows XP
- Unix fra ca. 1970 – mest *kommando drevet*, men også mange muligheter til pek-og-klikk, finnes i minst 15 varianter, men på Ifi:
 - Solaris
 - Linux på studentmaskinene
- Kjører du på det ene systemet, finnes det et program slik at du kan kjøre det andre
 - 'Windows oppå Linux' med kommandoen: `>windows`
 - eller 'Unix oppå Windows' med programmet X-Win32

30

Sett deg ned ved en ledig maskin

- Skjermen på en ledig datamaskin, klar til bruk

Institutt for informatikk - eina.ifi.uio.no:0

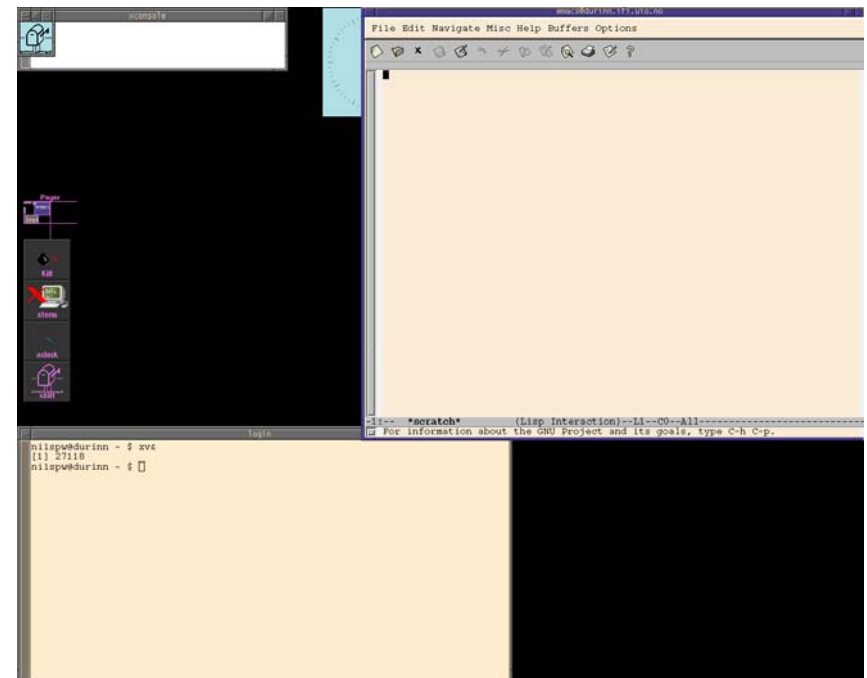
har.ifi.uio.no login: |

Password: ×

Tast inn ditt brukernavn og passord

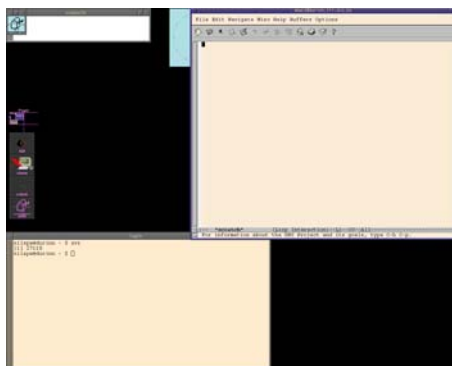
- Helt nøyaktig !
- Tast returtast  etter navn og etter passord

31



Skjermbildet etter innlogging:

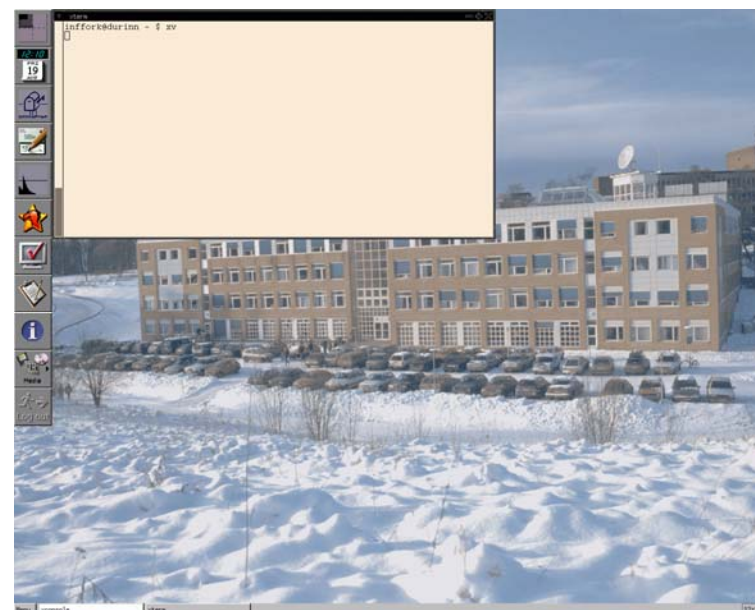
- Et Unix-vindu nederst til venstre (Xterm-vindu)
 - til å gi kommandoer
- Et emacs-vindu øverst til høyre
 - til tekstbehandling
- En markør - kan flyttes omkring på skjermen med musa
- Et lite panel – for kall av 4 programmer



Gå (klikk på) Unix-vinduet og gi kommandoen **installsetup** (og trykk Enter-tasten)

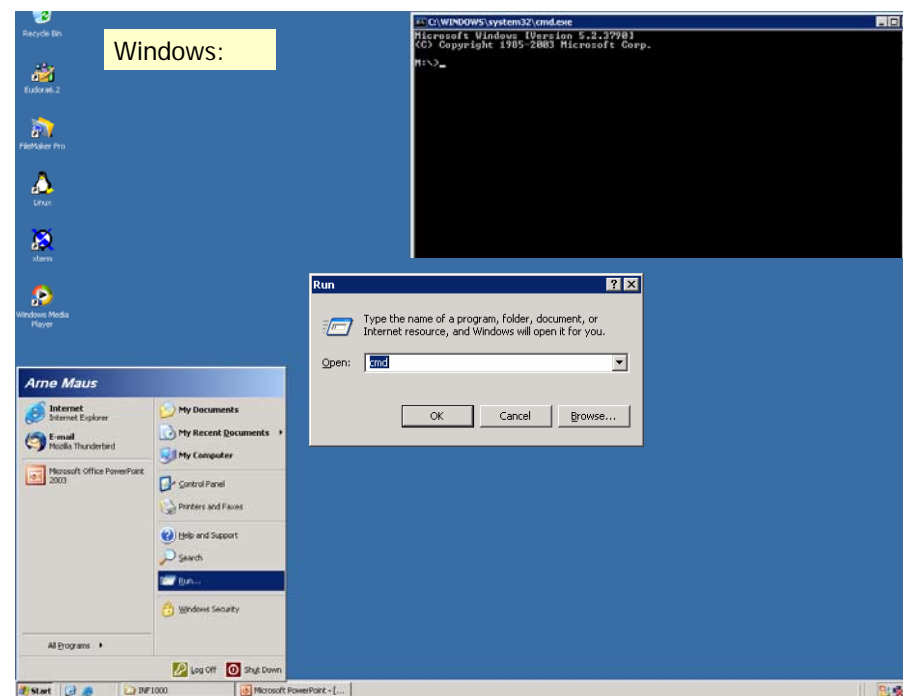
33

Etter **installsetup** får du standard Ifi oppsett (bedre) – også neste gang



Noe mer om manøvrering på skjermen

- 4 desk-top'er (=skjermflater) i Linux
 - Beveger du deg utenfor (ned og/eller til høyre får du ett annet (se øverst på den stående menyen om hvor du er)
- Høyreklikking av mus gir en meny (som du evt. må venstre-klikke i)
- diverse ulikheter Windows og Linux:
 - Linux gir tastetrykkene til det vinduet som musa er inni – Winws til det som har farget ramme
 - CTRL alt-del (Windows) mot ctrl D i Linux
 - Kommando-vindu i Windows (Start | Run og si: **cmd** som programnavn) – tilsvarende Xterm-vinduet i Linux
- Vi kan si at hver av Windows/Linux har sitt 'språk'



35

Flytt markøren til Unix-vinduet og tast:

```
einn: ~> hei
hei: Command not found.
einn: ~> hello
hello: Command not found.
einn ~> hjelp
```

Det finnes brukerhjelp for følgende områder:

```
applix      diskkvoter  ijk         oppringt    staroffice
backup      drift        ijk2        oppsett     studentby
....
disk-kvoter INF-kode    maskiner    simula
diskett     ifi         modem       skrivere
```

Ønsker du hjelp om andre emner enn de som er listet opp her, prøv kommandoen «man emne». For å skrive ut en fil, brukes opsjonen -P«printer».

Kommandoen «hjelp hjelp» gir full beskrivelse av dette.

```
einn: ~> hjelp infkode

/local/help/INF-kode.txt
.....
```

Hvordan få noe til å bli ?

Hvordan LAGRE noe til senere bruk ?

Løsning: Bruk **filer**

- En **fil** er en samling med data som oppbevares en stund i en datamaskin

Eksempler på filer:

- et brev
- en tabell med måleresultater
- en telefonkatalog

En 'stund':

- Fra noen sekunder til mange år

Hvor lagres det:

- magnetisk platelager (disken)

- Eksempel på en fil:

```
Kjære mor,
Jeg har det bra,
men Lånekassa er vanskelig.
Send mer penger!
Din Olemann
```

- Innholdet i denne fila (og andre) er tekst:
 - Teksten er delt i linjer
 - Linjene består av tegn.
- Hva slags tegn kan forekomme i en fil ? de viktigste:

```
ABC..XYZabcd..xyz ÅÖÅæå
01234...9
+\.,;:;!%&/()=#
```

- På våre datamaskiner (og de fleste andre):
 - 191 ulike tegn ('alle' går over til 'mange tusen')
 - 65 spesialtegn(f.eks "ny linje")
 - Hvert tegn er *kodet* internt med 8 bit (eller 16 bit i Java)

Om filer:

- Hva kan man gjøre med filer ?
 - Finne filer og vise fram innholdet
 - Skrive på papir ("printe") filer
 - Kopiere filer
 - Lage nye filer
 - Endre innholdet i filer (og legge til)
 - Fjerne filer
 - Lage områder for samhørende filer (kataloger) og flytte filer
- Noe av dette :
 - Finne, vise fram, skrive på papir, kopiere, fjerne, lage kataloger, flytte
 - gjøres med Unix-kommandoer
- Annet :
 - Lagre, endre gjøres med emacs, som er et tekstbehandlingsprogram

emacs - mange måter å gi kommandoer

- 1. Velg på menyene
 - klikk med musa
- 2. Trykk F10 -tasten
 - da deler vinduet seg, og man velger
 - først meny og så
 - funksjon (i den valgte menyen)
 - med piltastene og så returtasten CR
- 3. Med spesielle tastetrykk og funksjonstastene F1, F2, ...
 - mer nå og senere
- 4. Nederst på kommandolinja
 - for 'spesialister'

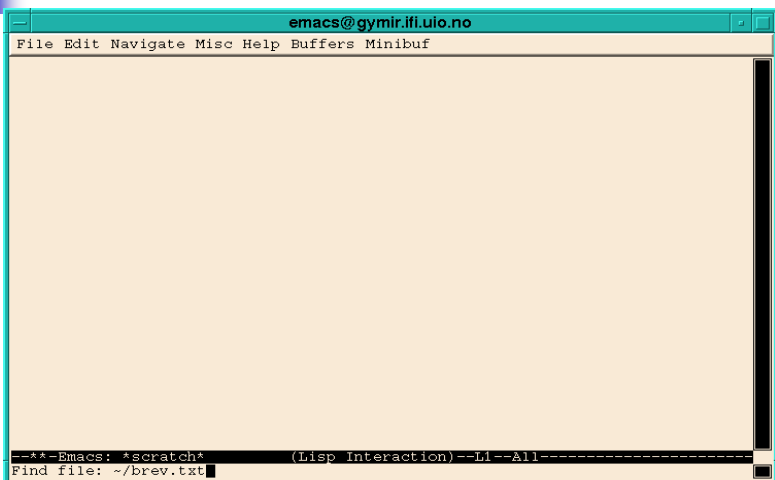
41

emacs

- Hvordan lage en ny fil i ved hjelp av emacs:
 - Flytt markøren inn i emacs-vinduet
 - trykk F3-tasten
 - Skriv navn på filen, f.eks brev.txt
 - trykk returtasten ↵
 - Skriv teksten som skal være i filen
 - Trykk F4-tasten, trykk y-tasten (data du skrev, lagres nå)

42

Lage en ny fil i emacs



43

Inntasting av data til 'brev.txt'



Husk å lagre filen

- trykk F4 eller
- velg 'save-buffer' fra 'File' menyen

44

Framvising og 'printing' av fil

Flytt markøren tilbake til Unix-vinduet og tast kommandoer:

```
einn: ~> ls
brev.txt
einn: ~> more brev.txt
Kjære mor,
Jeg har det bra,
men Lånekassa er vanskelig
Send mer penger!
Din Olemann
einn: ~> print brev.txt
print: Queuing text file brev.txt on lucida [1 sheet]
```

Tre nye - og nyttige-kommandoer ble brukt:

```
ls          - vis filnavnene på filområdet
more        - skriv innholdet av en fil sidevis
print       - skriv ut innholdet i en fil på 'printer'
```

Filområder

- Hver bruker har sitt eget filområde (katalog, mappe) med god plass til egne filer
- Alle filer har navn. Filene på et område må ha ulike navn.
- Ditt filområde:
 - Heter: ~dittbrukernavn
 - Det som står først i område-navnet, (altså: ~), er en såkalt tilde eller krøll-strek (tast først AltGr~, så mellomrom)
- Kommando for å få oversikt
 - ls områdenavn
 - lager oversikt over navnene til filene på området
- Kommando for å se innholdet i en fil
 - more filnavn
 - viser fram innholdet i filen, en side ad gangen.

46

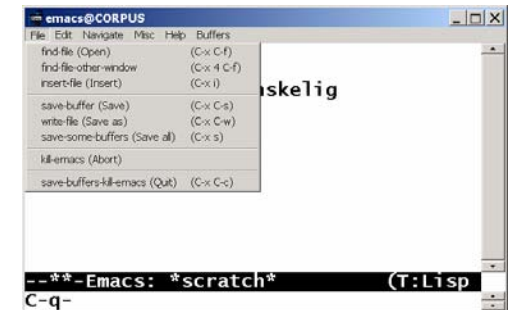
Litt flere kommandoer i Unix

```
einn: ~>copy ~inf1000/README .
einn: ~>ls
brev.txt    README
einn: ~> del README
einn: ~> ls
brev.txt
einn: ~> cd ~inf1000
einn: inf1000> ls
Arkiv      README  bin    eksamener  programmer  www_docs
Arkiv.old  V04    data  grupper    tmp
einn: inf1000> del README
del: README is write-protected;
    use `chmod' to change protection
    first.
einn: ~>windows
```

Denne siste kommandoen gir deg et vanlig Windowsgrensesnitt under Unix !

Hvordan avslutte (logge ut)

- 1) Lagre alt du har skrevet og endret i emacs
 - velg i fil-menyen: **save-buffers-kill-emacs(Quit)**
 - eller: trykk F4
- 2) Gå til 'xterm-vinduet'
 - skriv: **exit** og CR



48