

## Laborationer i kursmomentet Datoranvändning E1

<http://www.etek.chalmers.se/~hallgren/Eda/>

Laboration nr 1 : Vi inleder vår resa ut i datorvärlden

Laboration nr 2 : UNIX, e-post och nyhetsgrupper

Laboration nr 3 : Matematikverktyget Maple

Laboration nr 4 : Dokumentproduktion med FrameMaker

Laboration nr 5 : Mer om FrameMaker

Laboration nr 6 : Mer om UNIX

Laboration nr 7 : Programmering i Maple

Laboration nr 8 : Matteverktyget MATLAB

Laboration nr 9 : Textredigering, Unix och praktik

Laboration nr 10 : Att göra webbsidor

# Laborationer i kursmomentet Datoranvändning E1

<http://www.etek.chalmers.se/~hallgren/Eda/>

## Laboration nr 1: Vi inleder vår resa ut i datorvärlden

Originaltext: Olle Lundh

Omarbetning 95-08-21, 96-08-23, 97-08-20: Magnus Bondesson

Uppdatering 1998 och 99-09-21: Thomas Hallgren

### 1 Introduktion

Syftet med denna första laboration vid datorn är framför allt att vi skall bekanta oss med den praktiska hanteringen av E:s arbetsstationer och förhoppningsvis uppleva dem som goda kamrater. Vi passar samtidigt på att lära oss använda ett par nyttiga program.

#### 1.1 Förberedelser

Innan du kommer till laboration 1 är det lämpligt att ha läst eller skummat följande avsnitt i Gula Boken: Introduktion (avsnitt 1.7-1.8.1, 1.12 och 1.17.4), Fönstersystemet X (kapitel 2), WWW (5.3). Det är inte lätt att läsa om sådant här om man inte har haft praktisk kontakt med UNIX-maskiner, så hoppa bara över det bland dessa sidor du tycker är konstigt och återvänd i stället efter laborationen.

#### 1.2 Redovisning

För att bli godkänd på kursen måste man delta i laborationerna. Därför är det normalt att när man är klar med en laboration visa vad man har gjort för en handledare och bli godkänd av honom/henne.

### 2 Uppgifter

Alla laborationer bygger på att du arbetar och provar dig fram självständigt. Ha Gula Boken till hjälp. Om du fastnar eller om det är något som du inte förstår så fråga handledaren. Sitt inte för länge och grunna på något du inte blir klok på. Texten i uppgifterna innehåller en del material från första föreläsningen, men det förutsättes nog att du deltagit i den.

Symbolen  markerar i uppgifter med flera delar inledningen på ett stycke med en konkret arbetsuppgift. Dessförinnan kan finnas motiverande eller förklarande text. Anmärkningarna (inleds med **Anm.**) kan genomgående överhoppas om du inte är specialintresserad.

### Uppgift 1. Fönster och menyer

- Logga in på gruppens konto för denna kurs. Om det dyker upp ett informationsfönster, läs det och klicka bort det. Du bör nu ha minst ett sk terminalfönster i vilket du kan skriva UNIX-kommandon. De flesta kommandon resulterar i att ett program startas och utför den begärda uppgiften. Du skriver kommandot och avslutar det med retur-tangenten (↵). När fönstret är redo att ta emot ett kommando skrivs det ut en kort led-text, en sk prompt (sv redotecken), t ex

```
quarl14:~$ cd intro42/
```

där `quarl14` är datorns namn, `intro42` användarnamnet och placering i filsystemet anges inom klammrarna. `~` står för hemkatalogen.

Innan du kan mata in text från tangentbordet till ett fönster måste fönstret göras aktivt. Det du skriver på tangentbordet hamnar i det aktiva fönstret. Du gör ett terminalfönster aktivt genom att flytta muspekaren in i fönstret. Ramen på fönstret ändrar då färg.

- Prova följande kommandon:

```
date  
more1 /u1/ext/hallgren/Intro/Berling.txt
```

I det andra fallet ser du en sida i taget av en längre välkänd (?) svensk text. Tryck på mellanslagstangenten för att övergå till ny sida. Notera också att du med hissen (inom vissa gränser) kan komma tillbaka till sådant som redan visats i fönstret.

- Undersök vad som händer då du med muspekaren i kombination med musknapparna pekar, klickar och drar på ett fönsters ram och de figurer som finns i ramens övre del. Du kan på detta sätt flytta fönster, ändra fönsterstorleken i sidled och/eller höjddled, stänga fönster etc. Prova de olika alternativen.

Näm två sätt att flytta ett fönster: \_\_\_\_\_

Näm två sätt att ändra storleken på ett fönster: \_\_\_\_\_

Hur ikonifierar/deikonifierar man ett fönster? \_\_\_\_\_

- Placera muspekaren på området utanför fönstren på skärmen, tryck sedan ned de olika musknapparna och titta på de menyer som dyker upp (dessa menyer kallas **bakgrundsmeny**). Vänster musknapp ger en meny där du bland annat kan logga ut (kommandot **Quit/Logout**; detta använder du alltså när ditt datorpass skall avslutas) och skifta mellan engelsk och svensk teckenuppsättning på tangentbordet (detta skall numera vara onödigt, eftersom standardinställningen **Keyboard Options/Key Swe Ext** ger tillgång till alla tecken), mittenknappen ger en meny där du kan starta några olika typer av terminalfönster, högerknappen ger en meny där du kan starta några av de mest använda programmen. För att välja något i menyerna tar du fram aktuell meny och håller därefter musknappen nedtryckt samtidigt som du drar muspekaren på skärmen nedåt till önskat menyalternativ och släpper sedan musknappen.

---

1. Gissa varför kommandot heter `more`? Förresten verkar `more` visa ÅÄÖ riktigt. Det finns ett nyare kommando som heter `less`. Prova det istället.

Om du inte redan har två terminalfönster, så skapa ett nytt med valet *Xterm* i mittknappsmenyn.

Terminalfönster (som kan startas via bakgrundsmeny enligt stycket ovan) emulerar (dvs fungerar som eller liknar) gammaldags textterminaler. Du kan ha flera terminalfönster igång samtidigt och arbete kan ske i dessa oberoende av varandra, dvs skenbart har du tillgång till många datorer samtidigt. Vi skall inte illustrera detta närmare just nu.

- Lägg fönstren så att de delvis överlappar. Hur kan man få det undre fönstret att hamna överst (det räcker att någon del av ramen är synlig)? \_\_\_\_\_
- Man kan i allmänhet kopiera åtminstone text mellan fönster. Markera genom dragning med musen (vänstra musknappen nedtryckt) ett kommando i det ena fönstret och klistra in det i det andra (mellersta musknappen; sker på den lilla svarta rutans plats).

### Uppgift 2. Prova ett enkelt program

Starta med programmenyn (högerknappsmenyn) programmet *Calculator* och använd det. Det här programmet är ibland behändigt att ta till om man snabbt vill räkna ut något. Sedan tvingas vi fundera på hur programmet avslutas. Många program har en egen menyrad med kommando för stängning men så är det inte här, varför vi får använda menyknappen längst upp till vänster för att stänga fönstret och därmed programmet. Det finns ett sätt till om man är utrustad med bra minne: placera muspekaren i fönstret och tryck på tangenten `q` (av *quit*).

**Anm.** Programmenyns *HP-Calculator* startar en variant. De båda versioner kan alternativt startas med UNIX-kommandot `xcalc -stipple -geo 300x400` respektive `xcalc -rpn -stipple -geo 400x250`. Det är alltså samma program, som startas med olika parametrar. Det finns också en speciell Digital-version som startas med `dxcalc`.

### Uppgift 3. Prova ett annat enkelt program

- Starta med UNIX-kommandot `xpaint` ett ritprogram. Programmet är lättanvänt utan ingående instruktion och småroligt, men du kommer nog inte att ha så stor användning av det. Börja med att skapa en ritarea med **New Canvas** i **File**-menyn. Prova bl a kopiering/klistring med **Edit**-menyns kommandon. Programmet kan öppna och spara bilder, vilket sker på samma sätt som i kommande program, så strunta i det nu. Programmet har en egen menyrad och du avslutar det med **Exit** under **File**. Lägg märke till att programmet håller reda på att du gjort ändringar och ger dig en chans att spara innan det avbryts för gott. Bry dig dock inte om det just nu (filerna blir stora) utan besvara frågan med **No**.

**Anm.** Programmenyns *Paint* startar ett liknande men svagare program kallat *dxpaint*.

- Starta samma program igen i avsikt att för kommande behov göra en liten bild. Skapa nu ritarean i stället med **New With Size** och välj storleken till 100 x 50. Rita bilden och spara den med **Save As** och under GIF-format som filen `MINBILD.gif`. Gå sedan ur med **Exit**. Skriv sedan i terminalfönstret `xv MINBILD.gif` för att kontrollera att

den existerar (`xv` startar ett bildvisningsprogram). Avsluta det genom att placera muspekaren i bilden och trycka på tangenten `q`.

#### Uppgift 4. Kommandotolken `tcsh`

När man startar ett terminalfönster startas ett program kallat kommandotolk (eng. shell) som tolkar och utför de kommandon som du skriver. Den kommandotolk vi använder på E-linjens Unix-datorer heter `tcsh` (utläst T-C-SHELL).

En trevlig finess med `tcsh` är att den kommer ihåg kommandon som du matar in, så du behöver inte skriva jobbiga och långa kommandon mer än en gång. Du kommer åt de kommandon som du tidigare skrivit med hjälp av tangentbordets uppåtpil. Med nedåt- och uppåtpiltangenterna kan du bläddra fram och tillbaka bland de tidigare kommandona. Du kan också redigera kommandona med hjälp av höger- och vänsterpiltangenterna och på så sätt t ex. rätta kommandon som skrivits in felaktigt.

- Aktivera det första terminalfönstret och prova (börja t ex. med det felaktiga kommandot `datum` som sedan korrigeras till `date` utan omskrivning av de tre första bokstäverna). Verifiera också att `more`-kommandot från uppgift 1 finns kvar.

#### Uppgift 5. Kommandotolken `tcsh` igen

`tcsh` har viss förståelse för hur kommandon och filnamn, som är vanliga kommandoargument, ser ut. Om du trycker på `<TAB>` när du börjat skriva ett långt kommando eller filnamn kommer `tcsh` att försöka fylla i resten av namnet åt dig. `tcsh` kommer att fylla i så många bokstäver framåt som kan matchas unikt mot tillgängliga kommandon och filnamn. Om t ex flera filnamn börjar på samma sätt måste du hjälpa till genom att fylla i tecken som skiljer dem åt. Du kan därefter trycka på `<TAB>` igen för att få resten av namnet ifyllt.

- Prova med att t ex skriva `da` i ett terminalfönster och tryck därefter på `<TAB>` så fyller `tcsh` i resten. Vilket kommando blev det? \_\_\_\_\_

- `<Ctrl-D>` kan användas för att se de alternativ som återstår när du skrivit delar av t ex ett kommando eller en sökväg i systemets filhierarki. Skriv ett `x` på en tom rad och håll därefter CTRL-tangenten nedtryckt samtidigt som du trycker på D-tangenten (= `<Ctrl-D>`). Programmet `tcsh` listar då samtliga tillgängliga kommandon som börjar på bokstaven `x`. Fortsätt med att efter `x`:et skriva ett `c` och tryck sedan `<Ctrl-D>`. Du får nu en lista på de kommandon som börjar på `xc`. Fortsätt med ett `a` (dvs `xca`) och `<Ctrl-D>`. Som du ser finns det bara ett program som börjar med `xca`. För att skriva ut hela namnet behöver du nu bara trycka på `<TAB>` så fyller `tcsh` i det sista. Starta sedan programmet genom att trycka på Return-tangenten. Hur många program börjar på bokstäverna `xc`? \_\_\_\_\_

#### Uppgift 6. Filsystemet och några UNIX-kommandon

Som du förhoppningsvis kommer ihåg från föreläsningen ligger program och datafiler i UNIX (och de flesta andra populära operativsystem) ordnade i ett hierarkiskt filsystem. T ex har varje nybliven E1:a en egen katalog (eng. directory) kallad hemkatalogen i katalogen `/u1/e99`. Du kan alltid ta reda på vilken den aktuella katalogen är

med kommandot `pwd` (print working directory). Vid inloggning hamnar man i hemkatalogen, dvs den aktuella katalogen är hemkatalogen.

- Vad är det fullständiga namnet på labgruppens hemkatalog? \_\_\_\_\_

Man kopierar filer med kommandot `cp` (av copy) enligt

```
cp filnamn kopian_filnamn
```

- Skapa en kopia av filen `/u1/ext/hallgren/Intro/Berling.txt` hos dig och kalla den `Berling.txt`. Kommandot blir alltså:

```
cp /u1/ext/hallgren/Intro/Berling.txt Berlin.txt
```

Det är ganska långt, så det kan löna sig att använda `<TAB>`, som i förra uppgiften, för att få *tsch* att fylla i en del automatiskt.

OBS! I Unix är de flesta tecken tillåtna i filnamn, men vissa program kan vara kräsna. Det kan vara bra att undvika filnamn som innehåller mellanslag eller de svenska bokstäverna å, ä, ö, Å, Ä, Ö).

Hur kontrollerar man att kopian verkligen har skapats? Kommandot `ls` i terminalfönstret skriver ut vilka filer som finns i den aktuella katalogen. Om du lägger till flaggan `-l` (dvs bokstaven lilla L, inte siffran 1) till kommandot `ls`, dvs skriver `ls -l` så får du lite mer information om respektive fil, bl a står när innehållet i varje fil senast ändrades, vem som äger filen och hur stor filen är.

- Tag nu en kopia av din egen fil och kalla kopian `Berling2.txt`. Kontrollera med `ls -l` igen. Byt sedan namn på denna andra kopia med `mv`-kommandot (av move) enligt `mv filnamn nytt_filnamn` till `Selma.txt`. Kontrollera igen med `ls -l` (använd uppåtpiltangent om du inte orkar skriva `ls -l` igen).

- Tag slutligen bort `Selma.txt` med `rm`-kommandot (av remove). Kontrollera.

Det är för framtida behov bra att veta att `..` syftar på katalogen ovanför den aktuella och att `.` avser den aktuella. Dessa beteckningar används bl a i vissa dialogrutor som vi snart möter.

- Tag med `ls -l ..` reda på vilka filer som finns i katalogen ovanför din.

### Uppgift 7. Pyssla med redigeringsprogrammet NEdit

*NEdit* är nog det program som majoriteten kommer att använda för redigering av rena textfiler, vilket man har nytta av i diverse sammanhang. Låt oss därför nu prova på några möjligheter. Starta redigeringsprogrammet *NEdit* (från programmenyn eller med kommandot `nedit` i ett terminalfönster). Öppna din fil `Berling.txt` (se uppgift 6) genom att välja **File/Open** i menyn.

- Gå till en viss rad i filen. Leta efter någon textföljd (t ex präst eller sup) i filen. Skriv in egen text. Kopiera/klistra på vanligt X-sätt, dvs med vänstra/mellersta musknappen, eller med menykommando. Testa ångra-kommandot **Undo** som i detta program minns flera steg bakåt. Spara under nytt namn i din hemkatalog.

**Anm.** Ett annat populärt textredigeringsprogram, som beskrivs utförligt i Gula Boken, är *Emacs*. Det startas med kommandot `emacs` i ett terminalfönster. Prova det också när du har tid.

### Uppgift 8. Surfa på nätet

Man kan utföra väldigt mycket vid en dator utan att behöva bekymra sig om vanliga UNIX-kommandon. Det har vi redan sett små exempel på i uppgifterna 2 och 3. Vi skall nu övergå till att titta litet på ett modernt kraftfullt program med hyggligt användargränssnitt. Vi väljer webbläsaren (eng. web browser) *Netscape*.

**Anm.** Beskrivningen gäller version 3 av webbläsaren. Version 4, var inte tillgänglig på E-sektionens datorer då detta skrevs. Den största konkurrenten *Internet Explorer* finns bara för vissa UNIX-varianter.

- Starta *Netscape* från programmenyn. Det dyker nog upp en sida med engelsk text. Klicka med musen någonstans på sidan och strax kommer du till E:s hemsida och kan fortsätta din vandring ut i världen. Det kan vara värt att notera att adressen (webb-adressen) - eller referensen - till det dokument du betraktar anges i övre delen av fönstret med **Location**: <http://www.etek.chalmers.se/>
- Minska höjden på fönstret och notera att du då får en hiss med vilken du kan glida över dokumentet. Prova och återgå sedan till ungefär samma storlek som från början.
- Ge dig nu ut på en vandring på egen hand. Notera att du med **Back**- och **Forward**-knapparna kan vandra fram och tillbaka mellan redan besökta sidor. Observera också att du med **Bookmark**-menyn kan spara adresser till intressanta sidor för snabbval vid senare tillfälle. Spara några adresser i den och testa. Med **Go**-menyn kan du hoppa mellan redan besökta sidor.

### Uppgift 9. Besök kursens hemsida

Adressen till kursens hemsida är

<http://www.etek.chalmers.se/~hallgren/Eda/>

- Besök den! Antagligen hittar du ingen länk till den, så du får själv mata in adressen i **Location**-rutan. Man kan oftast hoppa över `http://` när man matar in adresser.
- Sätt ett bokmärke på sidan! Det kan vara bra att ha, eftersom det är det lite dåligt med länkar till kursinformation från E:s hemsida.
- Hur många av lärarna på kursen har rörliga bilder på sin hemsida? \_\_\_\_\_
- Leta upp länkten till Gula Boken på webben och klicka på den. Boken kommer att visas av programmet *Acrobat Reader*, som startas automatiskt. Notera att man kan klicka i innehållsförteckningen och indexet för att hoppa till olika avsnitt i boken och bläddra mellan sidorna med tangenterna Page Up och Page Down.
- Under Länkar på kursens hemsida finns en länk till sidan *Länksammanställning från Lab-PM:en*. Ta en titt på den!

### Uppgift 10. WWW-kataloger och sökmaskiner (söktjänster)

När man letar efter information finns det flera vägar att gå. Många personliga sidor innehåller utvalda länkar för ett eller flera intresseområden. Det ligger säkert en hel del möda bakom dessa sidor.

För seriöst arbete kan man ta sig till ett par ställen som innehåller strukturellt ordnade kataloger med adresser. Med knappen **Länkar** på E:s hemsida kommer man till en sida med länkar till bland annat ett antal kataloger över WWW-resurser. Exempel på sådana är (inom parentes anges motsvarande direktadress) **SUNETs** katalog (<http://www.sunet.se/sweden/main-sv.html>; man hamnar i katalogen över svenska WWW-resurser; SUNETs hemsida är <http://www.sunet.se>), **YAHOOs** internationella och svenska katalog (<http://www.yahoo.com> resp <http://www.yahoo.se>), samt Open Directory Project (<http://dmoz.org>). Om du vill få en uppfattning om ökningen av registrerade sidor i Sverige kan du trycka på knappen **Nyheter** i SUNETs katalog.

Både SUNET och Yahoo har sökfunktioner som man kan använda istället för att bläddra sig fram i kategorierna för att hitta det man söker. En annan typ av söktjänst är t ex *Evreka* och *HotBot* (<http://www.evreka.com> resp <http://www.hotbot.com>) som också dessa kan nås t ex via E:s länksida.

SUNETs och Yahoos sökningar är mera begränsade än Evrekas och HotBots, genom att de huvudsakligen söker i katalogen över registrerade sidor. Evreka har samlat på sig en massa information om webbsidor i hela världen och söker alltså i denna information. Eftersom webben ständigt förändras är informationen aldrig helt aktuell, och de svar man får hänvisar därför ibland till webbsidor som är borttagna eller inte längre innehåller den sökta informationen. Detta gäller tyvärr både katalogerna och söktjänsterna.

Sättet att söka i de olika söktjänsterna är också litet olika. Om man skriver in många ord betyder det oftast att man får en lista över webbsidor där minst ett av orden finns med, sorterade så att de sidor där alla orden finns med kommer först. Titta på hjälpsidorna för respektive sökmaskin för detaljer. Ofta kan man välja mellan enkel eller avancerad sökning.

- Tag dig på detta sätt till YAHOO och begrundade hur många programspråk det finns information om eller ge dig på något annat ämne av intresse.
- Besök även AltaVista eller HotBot och sök efter något av intresse.
- Besök svenska Yahoo eller SUNETs katalog över svenska WWW-resurser och se efter vad det finns för material om en svensk kommun som intresserar dig.

### Uppgift 11. Sökning efter information i en kommersiell databas

Allt som finns på webben går inte att hitta via de generella tjänsterna vi sett ovan. Den information som finns där är den man kan hitta genom att bara klicka sig fram.

Det finns ett växande antal fria och kommersiella databaser som kan nås via WWW. En fri sådan är *Internet Movie Database* (<http://www.imdb.com>), där man kan slå upp information om filmer och skådespelare, mm. En kommersiell sådan är *Britannica Online*, som baseras på uppslagsverket *Encyclopedia Britannica*. Chalmers är abon-



ment på denna databas och du kan därför nå den utan lösenord el dyl från arbetsstationerna på Chalmers<sup>1</sup>. Adressen är

<http://www.eb.com:180/>

(Om man går via länken på E:s hemsida (**Länkar - Britannica Online**) kommer man till en gratisversion där man endast kan se smakprov ur uppslagsverket.)

- För min generation är namnen Louis Armstrong (jazz-musiker) och Neil Armstrong (den första människan på månen) välkända och finns i alla uppslagsböcker. Det finns också en person med detta efternamn som haft stor betydelse för radiokommunikation och elektronik. Slarvigt och egenartat nog finns han dock inte nämnd i vår svenska Nationalencyklopedi, men ägnas åtskilliga rader i Encyclopedia Britannica. Tag med hjälp av Britannica Online, reda på

- a) Hans båda förnamn \_\_\_\_\_
- b) Vilken kommunikationsteknik uppfann och kämpade han för? \_\_\_\_\_
- c) Vad fanns dessförinnan (två bokstäver)? \_\_\_\_\_
- d) Hur dog han och vilket år? \_\_\_\_\_

Däremot letar man förgäves efter t ex barnboksförfattare som Enid Blyton, Edward Ellis och W.E.Johns men hittar Astrid Lindgren.

### Uppgift 12. Leta efter böcker på biblioteket

- Gå till Chalmers biblioteks hemsida:

<http://www.lib.chalmers.se>

Via den (det finns andra sätt) kan man leta efter böcker (och även reservera böcker om man har lånekort).

- Välj **CHANS** (bibliotekets katalog) på bibliotekssidan. Då startar ett program som visar upp ett fönster. Detta söksystem är ganska gammalt och inte anpassat till webben, men är nog lättare att använda än det kan se ut vid en första anblick. Följ anvisningarna (ange användarnamnet CHANS först). Undersök t ex om det finns en bok av Richard Murray som handlar om robotteknik och huruvida den är utlånad eller inte. Naturligtvis kan du välja något annat namn av större intresse för dig.

### Uppgift 13. Enkät

- Det finns många användningsområden för WWW. Ett är att samla in information från användare, kunder etc. För att vi skall få se ett exempel på detta har jag gjort i ordning en rätt oskyldig enkät. Denna hittar man om man går till kursens hemsida (tips: använd bokmärket du gjorde i uppgift 9!), klickar fram fliken **Datorövningar** och klickar på länken **Enkät** vid laboration 1 i tabellen. Låt varje gruppmedlem besvara den. Eventuellt kommer vi i läsperiod 2 att säga något om vad som ligger bakom den teknik som används i detta sammanhang.

---

1. Man kan även nå den från sin hemdator om den är uppkopplad via Chalmers.

## 3 Frivilliga extrauppgifter

### Uppgift 14. Fortsatt nätsurfning

De flesta har kanske redan surfat på nätet en del, men för den som inte har det kanske följande kan vara nyttigt.

Man kan ta sig till ett godtyckligt dokument genom att i **Location**-rutan skriva in en adress (klicka först med musen där) och sedan trycka på returknappen.



Tag dig på detta vis till t ex någon av adresserna

<http://www.aftonbladet.se>

<http://www.gp.se>

<http://www.idg.se>

<http://www.nyteknik.se>

och se om du kan hitta någon god nyhet. Om du vid en dator som har ljud, tycker jag att du också bör passa på att besöka

<http://www.sr.se/ekot/nyheter/>

Välj EKOT och lyssna t ex på en del av senaste radioekot.

På adressen

<http://www.skolverket.se/skolnet/smultron/>

finns en utförlig introduktion till Internet och WWW. Klicka dig fram till avsnitt 2.9 och titta på listan med exempel på vad WWW används till idag.

Några andra adresser som antyder rikedom i WWW:

<http://www.lysator.liu.se/runeberg/>

Elektronisk litteratur

<http://www.sverigeturism.se/index.htm>

Turism

<http://www.imdb.com>

Filmdatabasen

<http://www.passagen.se>

Telias marknadsplats

<http://www.torget.se>

Postens motsvarighet

Som redan nämnts kan du spara intressanta adresser som s k bokmärken med Book-mark-menyn.

## 4 Vill du veta mer?

Via kursens hemsida

<http://www.etek.chalmers.se/~hallgren/Eda/>

kan du hitta länkar och annat material som har med föreläsningarna/laborationerna att göra. Nytt stoff läggs till successivt.

Länkar som förekommit i laborationerna finns (som sagt) sammanställda på sidan *Länksammanställning från Lab-PM:en*. Adressen till denna är:

<http://www.etek.chalmers.se/~hallgren/Eda/lablinks.html>

## 4.1 LITEN ORDLISTA

Logga in	att identifiera sig för datorsystemet.
Logga ut	att avsluta sitt arbetspass så att datorn är klar för nästa användare.
Terminalfönster	program som emulerar (efterliknar) en gammaldags textterminal.
Returtangent	↵-tangenten på tangentbordet (=return).
Default	det värde/den inställning som gäller då inget anges explicit.
WWW	World Wide Web.
HTML	Hyper Text Markup Language, det språk som beskriver de flesta av de dokument som Netscape och andra webbläsare gillar.
webb	försvenskad form av WWW (jfr jobb av eng job).
webbsida	allmän beteckning på ett dokument som läses via webben; kan vara en HTML-sida, bild, animering, eller ljudstycke.
hemsida	den webbsida som en person eller en organisation presenterar sig med och som utgör ingången för ytterligare information; kan alternativt betyda den sida som webbläsaren visar när man startar den.

## Laborationer i kursmomentet

### Datoranvändning E1

<http://www.etek.chalmers.se/~hallgren/Eda/>

## Laboration nr 2: UNIX, e-post och nyhetsgrupper

Originaltext: Olle Lundh

Omarbetning 95-08-21, 96-09-05, 97-09-03: Magnus Bondesson

Omarbetning 1998 och 99-10-26: Thomas Hallgren

## 1 Introduktion

Den här gången tittar vi lite mer på UNIX-kommandon som har med filer och filsystemet att göra. Ytterligare UNIX-material kommer någon gång i framtiden. Dessutom lär vi oss att använda e-post och nyhetsgrupper. **Arbeta hela tiden på kurskontot.**

### 1.1 Förberedelser

Innan du kommer till Laboration nr 2 är det lämpligt att ha läst eller skummat igenom uppgifterna och följande sidor i Gula Boken: Litet om allt (1.9-1.14), Dokumentation (kapitel 3), Program och kommandon (kapitel 4, skumma bara), e-post och nyheter (5.2 & 5.4, skumma bara). Hoppa över det bland dessa sidor du tycker är konstigt och återvänd i stället efter laborationen.

### 1.2 Redovisning

Kom ihåg att laborationerna obligatoriska moment i kursen. Labhandledarna håller koll på vilka grupper som är klara med vilka labbar, så prata med en handledare när ni är klara med en laboration. Han kanske ställer några frågor för att kolla att ni har gjort uppgifterna, så skriv ner svaren på de frågor som finns i anslutning till några av uppgifterna. Vissa uppgifter godkänns på annat sätt (t ex att man har skickat elektronisk post i Uppgift 4 i denna laboration).

## 2 Uppgifter

Som tidigare markerar symbolen  i uppgifter med flera delar inledningen på ett stycke med en konkret arbetsuppgift. Dessförinnan kan finnas motiverande eller förklarande text.

### Uppgift 1. Enkäten

Om du inte redan har besvarat enkäten (Uppgift 13 på Laboration nr 1), gör det nu.

## 2.1 Elektronisk post

Elektronisk post (datorpost, e-post, e-mail) handlar om att kunna skicka meddelanden per dator till en eller flera personer och kunna läsa sådana som andra personer skickat till dig.

I UNIX-system har alla användare normalt en egen brevlåda. Brevlådans adress är samma som kontonamnet, t ex `introxx` för de konton som används i denna kurs. Är UNIX-system anslutet till Internet kan man, förutom att skicka meddelanden till andra användare på samma system, även skicka meddelanden till och ta emot meddelanden från användare på andra Internet-anslutna datorsystem.

När man skickar meddelanden mellan olika datorsystem är adressernas utseende *användare@organisation*, där organisation normalt består av flera delar med punkt mellan. Tecknet @ uttalas som det engelska ordet *at* (ibland hör man också folk säga *snabel-a* eller *kanelbulle*). E-teknologer på Chalmers tillhör organisationen `etek.chalmers.se`.

Det finns flera program som man kan använda för att skicka och ta emot e-post. Vi tar en titt på hur e-post fungerar i *Netscape*.

### Uppgift 2. Signaturfil

Innan vi börjar skicka e-post-meddelanden utför vi en engångsåtgärd. Man brukar ju skriva under brev man skickar med sitt namn (och kanske något mer). För att slippa skriva under varje gång man skickar ett nytt brev kan man skapa en fil med namnet `.signature` i sin hemkatalog. Många e-post-program lägger då automatiskt till innehållet i denna fil sist i de brev man skickar.

- Skapa med *NEdit* (eller *Emacs*) i hemkatalogen `.signature` som innehåller era namn och labgruppens e-postadress.

### Uppgift 3. Läsa elektronisk post i Netscape

- Starta nu *Netscape* om programmet inte redan är igång. Om du inte redan lagt märke till det så uppmärksamma att man får hjälptexter om markören får vila en stund på en ikon. Välj menyalternativet **Window/Netscape Mail**, så dyker det upp ett postfönster (första gången visar sig först en dialogruta om att katalogen `nsmail` inte existerar; klicka bara på **OK**). Klicka på bilden **Inbox**, så får du se ett välkomstbrev från *Netscape*, men några andra brev finns nog inte.

Längst upp till höger på bildskärmen finns en postlåda. På den kan man se om det finns nya meddelanden. Även *Netscape* undersöker med jämna mellanrum om det finns nya meddelanden i brevlådan och ändrar utseende på indikatorn i nedre högra hörnet av fönstret. Om Mail-fönstret är ikonifierat ändrar ikonerna utseende.



- För att läsa nyinkomna meddelanden ska man trycka på knappen **Get Mail**. Gör det. Förhoppningsvis dyker ett meddelande från Thomas Hallgren upp i **Inbox**. Läs det.

#### Uppgift 4. Att svara på elektroniska brev

- Antagligen vill Thomas att ni skickar ett svar, så nu gör vi det. Tryck på knappen **Re: Mail** (Re som i reply). Vi får nu upp ett nytt brevskrivfönster, där adressaten (Mail To:) och rubriken (Subject:) redan är ifyllda. Vidare inkluderas det ursprungliga brevs text, vilket brukar vara bra när det gäller frågesammanhang, men du kan naturligtvis redigera bort delar. Skriv ditt svar (du kan redigera med piltangenter och musens hjälp, etc). Skicka iväg meddelandet genom att trycka på knappen **Send**.

#### Uppgift 5. Att skapa nya elektroniska brev och skicka dem

- Skicka e-post till den egna gruppen. För att få fram ett fönster att skriva brevet, klicka på knappen **To: Mail**. Fyll i adressen och rubriken (eng. subject) och skriv brevet på ett naturligt sätt. Om du vill skicka en kopia till någon kan du fylla i en adress till vid **Mail CC** (CC=Carbon Copy). Kontroll-läs brevet och tryck sedan bara på **Send**-knappen.
- Nu skall vi strax läsa brevet. När du skickat iväg brevet så vänta en stund till dess postlådan i övre högra hörnet av skärmen indikerar att du fått post (den reagerar dock långsamt så du kanske inte orkar vänta så länge). Tryck på knappen **Get Mail** igen, och notera att brevet dyker upp i **Inbox**. Notera också att du med musens hjälp kan vandra runt bland de brev du nu har i **Inbox**. En kopia av skickade brev hamnar för övrigt alltid i mappen **Sent**, som du också kan inspektera. Återvänd till det mottagna brevet i mappen **Inbox**.
- Skicka eventuellt e-post till någon som sitter vid en dator intill dig.

#### Uppgift 6. Att ta bort elektroniska brev

- Du kan ta bort ett brev, t ex genom att först markera det i förteckningen över brev och sedan trycka på **Delete**-knappen (i menyn!). Brevet flyttas då till mappen **Trash** (sv. soptunna, papperskorg). Där ligger det kvar till dess du använder menyalternativet **File/Empty Trash Folder**.

#### Andra program för elektronisk post

*Netscape* är ett stort program som kan ta lång tid att starta. Det finns dock andra e-post-program man kan använda, om man t ex bara snabbt vill kontrollera den senast anlända posten, eller skicka ett litet enkelt meddelande. Exempel på sådana är *elm*, *pine* och *mail*. De två förra beskrivs i Gula Boken. Gemensamt för alla tre är att de körs i ett terminalfönster. Detta gör att de kan passa bra att använda när man är inloggad på E-sektionens datorer via modem.

#### Varning för Netscape

Många program för e-post tillåter att man sorterar sin post i olika mappar. I de flesta program ligger meddelanden kvar i användarens brevlåda tills de flyttas till en annan map. *Netscape* skiljer sig dock på denna punkt: när man trycker på knappen **Get Mail** flyttas genast alla nyinkomna meddelanden bort från brevlådan till mappen som heter **Inbox**. Detta kan leda till två problem:

- När man har läst sin post med Netscape syns inte meddelandena längre i andra e-post-program.
- Om inte det lediga filutrymmet i användarens hemkatalog är tillräckligt stort för att lagra nyinkomna meddelanden, så kan man inte läsa dem med *Netscape*.<sup>1</sup>

I Uppgift 14 tittar vi på hur man kan hålla koll på det lediga filutrymmet och undvika det senare av dessa två problem.

## 2.2 Nyhetsgrupper

Nyhetsgrupper (även kallade diskussionsgrupper, eng Usenet news) är ytterligare ett sätt att sprida information via internet. Om du missade föreläsningen då jag pratade om detta, ta en titt på webbsidan

<http://www.skolverket.se/skolnet/smultron/news.htm>

som ger en kort introduktion och länkar till mer information.

Att läsa/skriva meddelanden i nyhetsgrupper görs på nästan samma som man läser/skriver e-post-meddelanden. Precis som det finns speciella program för e-post, finns det också speciella program för nyheter. *Netscape* går också bra att använda: menyalternativet **Windows/Netscape News** öppnar *Netscapes* nyhetsfönster.

### Uppgift 7. Läsa nyheter i förvalda grupper

- Öppna nyhetsfönstret i Netscape och notera att det dyker upp några nyhetergrupper upp till vänster. Det kanske är för trångt från början för att man ska se vad grupperna heter, men man kan dra i kolumnrubrikerna för att skapa mera plats. Det kan också hända att Netscape inte är rätt inställt från början och att listan därför är tom. Fråga i så fall handledaren om hjälp med att rätta till detta.)
- Det borde dyka upp en grupp som heter **news.announce.newusers**. Välj den och läs i meddelandet "Emily Postnews Answers Your Questions on Netiquette". Vad ska man göra, enligt Emily Postnews, om man har skickat ett meddelande och glömt skriva under?

---

Notera också att det finns ett meddelande med rubriken "A Primer on How to Work With the Usenet Community". Den kan vara bra att läsa innan man börjar skicka meddelanden till nyhetsgrupper.

### Uppgift 8. Att välja nyhetsgrupper

- I nyhetsfönstret syns normalt bara de nyhetsgrupper som man prenumererar på och som innehåller olästa meddelanden. I **Options**-menyn kan man ändra på det. Alternativet heter **Show ... Newsgroups**, där ... är **Subscribed**, **Active** eller **All**. Välj alternativet **Subscribed**. Vilka grupper relaterade till E-sektionen visas? \_\_\_\_\_

1. Jag vet inte hur bra felkontroll Netscape har i denna situation. Om det finns brister här finns alltså risken att nyinkomna meddelanden försvinner spårlöst...

Välj alternativet **Options/Show All Newsgroups**. Om inga E-relaterade grupper dök upp i förra uppgiften, leta upp dem här bocka för dem. De finns under **cth**. (Om inga grupper visas, fråga handledare om hjälp.) Välj också någon mer grupp (t ex **cth.dtek.newsoft** om du är nyfiken på vad D-sektionen installerar för program på sina datorer, eller **cth.vision-debatt** om du är nyfiken på vad folk tycker om Chalmers framtida utveckling).

### Uppgift 9. Uppgift i nyhetsmeddelande

- Läs meddelandena i de grupper som har med E1 att göra. Förhoppningsvis hittar du ett meddelande med ytterligare någon uppgift från Thomas Hallgren. Gör även denna uppgift.

### Uppgift 10. Avsluta Netscape

- Innan man loggar ut är det bra att avsluta *Netscape* med **File/Exit**. Om man inte avslutar *Netscape* innan man loggar ut kommer *Netscape* antagligen att klaga nästa gång det startas.

## Andra program för nyheter

Nyhetsgrupper har funnits mycket längre än webben, så det finns många program för att läsa nyheter. I Gula Boken beskrivs *knews*.

## 2.3 Filsystem

### Uppgift 11. Kataloger

Det är bra med ordning och reda i de flesta sammanhang. Eftersom du säkert på ditt personliga konto senare kommer att lägga mycket annat material, kan det vara bra ha särskilda *kataloger* (på engelska *directory*) för bl a olika kurser, t ex en katalog som har att göra med ditt projekt och en annan för matematik, osv. Kataloger skapar du med kommandot `mkdir` och du tar bort tomma kataloger med kommandot `rmdir`. I Windows 95 liksom MacOS använder man ordet *mapp* (på engelska *folder*) i stället för katalog och i Gula Boken kallas de bibliotek.

- Skapa en katalog *Datorintro* och flytta någon av gruppens filer dit. Flyttning gör vi ju med kommandot `mv källa destination` (se Uppgift 6 i Laboration nr 1). Källan skall vara ett filnamn och hittills har även destinationen varit det. Men den kan också vara en katalog, varvid det ursprungliga namnet bevaras. Hur gjorde du?

- 
- Med kommandot `pwd` (print working directory) kan du se var du befinner dig i filträdet (redotecknen visar det också). Kommandot `cd katalognamn` använder du för att förflytta dig ned i katalogen med namnet *katalognamn*. Flytta dig till *Datorintro* och kontrollera på sedvanligt sätt att filerna finns där. Kommandot `cd . .` (`cd`, mellanlag, två punkter) använder du för att gå ett steg upp i fil-trädet. Du kan alltid ta dig tillbaka till kontots hemkatalog med kommandot `cd` (utan *katalognamn*). Prova bägge sätten att från *Datorintro* återvända till hemkatalogen.



## Uppgift 12. Kopiering

Det händer att man vill kopiera mellan olika användare, t ex från ett kurskonto till ett personligt konto. Man ser då till att vara inloggad på det konto till vilket kopiering skall ske. Kopieringen måste gälla en fil som är läsbar (se nästa uppgift) för alla.

Med `~användarnamn` avser man hemkatalogen för en viss användare, vilket oftast är bekvämare än att skriva ut fullständigare namn. Tecknet `~` utläses "tilde". Tex avser `~intro130` och `/ul/kurs/intro/intro130` samma katalog och `~e9xxxx` samma som `/ul/e99/e9xxxx`.

Med `.` avser man aktuell katalog (`..` som vi mött tidigare avser katalogen ovanför).

Här ser du några varianter på en kopiering som du gjort tidigare:

```
cp ~hallgren/Intro/Berling.txt .
cp ~hallgren/Intro/Berling.txt Berling.txt
cp /ul/ext/hallgren/Intro/Berling.txt ~introxx
cp ~hallgren/Intro/Berling.txt B.txt
```

De tre första innebär samma sak (om den aktuella katalogen är gruppens hemkatalog), medan i den sista kopian får ett annat namn.



Kopiera en fil från någon av gruppmedlemmarnas personliga konton eller från `hallgren` till katalogen *Datorintro* och som skapades i Uppgift 11. Kontrollera.

## Uppgift 13. Filrättigheter

När du gjort i ordning en fil vore det i allmänhet inte så bra att vem som helst kunde ändra i den (inklusive ta bort den). Normalt har därför bara användaren (eng. user), dvs du själv, ändringsrätt - kallad **skrivrätt** - när det gäller filer och kataloger. Däremot har det i UNIX-världen, åtminstone bland akademiker, varit vanligt att man låter andra ha **läsrätt** till många filer. E-sektionens datorsystem har valt denna form av öppenhet, vilket betyder att filer normalt kan läsas av andra användare.

Kommandot `ls -l`, som vi sett förut, visar en massa information om filerna i en katalog, bland ann vilka rättigheter som gäller. Avsnitt 1.13 i Gula Boken beskriver detta.

Vi låter nu `hallgren` ta reda på information om alla filer i några kataloger (\* betyder noll eller flera godtyckliga andra tecken, dvs `KATA*` står för alla namn som inleds med `KATA`):

```
hortensia:hallgren:[~]> ls -l Intro Intro/KATA*
Intro:
-rw-r--r--  1 hallgren users      4491 Aug 19 17:42 DemoHTML.html
drwxr-xr-x  2 hallgren users      8192 Sep  2 16:24 KATALOG1
drwx-----  2 hallgren users      8192 Sep  2 16:24 KATALOG2
-rw-r--r--  1 hallgren users     3547 Aug 25 15:12 enkat1.html

Intro/KATALOG1:
total 2
-rw-r--r--  1 hallgren users       4 Sep  4 15:56 fill1
```

```
-rw----- 1 hallgren users      13 Sep  4 15:59 fil3
Intro/KATALOG2:
total 1
-rw-r--r-- 1 hallgren users      5 Sep  4 15:57 fil2
```

Filen `fil1` är läsbar för alla, medan `fil3` bara är läsbar för ägaren. Filen `fil2` är trots att det finns `r` rakt igenom inte läsbar för andra än ägaren, eftersom katalogen `KATALOG2` som innehåller filen inte är det.

- Försök att kopiera var och en av filerna `fil1`, `fil2` och `fil3` enligt ovan till din egen katalog. Katalogerna som innehåller filerna finns i katalogen `~hallgren/Intro/`. Förklara för dig själv.
- Flytta dig med `cd` till `KATALOG1` och gör `ls -l`. Flytta dig med `cd` till `KATALOG2`. Förklara för dig själv.

Nu uppstår frågan hur man ändrar rättigheterna, ty ibland vill man ju att andra inte skall ha läsrätt till ens filer. Man använder kommandot `chmod` (change mode) för detta. `Tex` betyder `chmod go-rw minfil` att filen `minfil` inte kan läsas eller skrivas av gruppen eller andra (+ inför rättighet). I enlighet med E-sektionens filosofi om öppenhet bör nog inte hemkatalogens rättigheter ändras, utan bara de för underliggande filer och kataloger. Vissa program, t ex `postprogram`, skapar kataloger som är lässkyddade från början.

- Ändra nu rättigheterna för någon av filerna i gruppens hemkatalog. Kontrollera resultatet med `ls -l`.

#### Uppgift 14. Hur vet man hur mycket ledigt utrymme det finns?

Även om hårddiskar blir större och billigare i ganska snabb takt är lagringskapaciteten hos dagens datorer begränsad. Att filsystemet är fullt är ett fel som inte alla programmerare tänker så mycket på när de skriver program. Ett välskrivet program kanske säger "filen kan inte sparas eftersom det inte finns tillräckligt med ledigt utrymme" då detta fel uppstår. Ett mindre bra program kanske säger "write error" och något felnummer. Ett ännu sämre program säger kanske ingenting utan sparar halva filen och fortsätter sedan som om inget hade hänt.

Det finns alltså anledning att hålla lite koll på hur mycket ledigt filutrymme man har. Kommandot `quota` ger information om hur mycket filutrymme den påloggade användaren får lov att använda.

```
hortensia:hallgren:[~]> quota
Disk quotas for user hallgren (uid 1503):
  Filesystem blocks  quota limit  grace  files quota limit grace
  /u1/ext       468  25000 30000      62    0     0
```

I exemplet ovan frågade användaren `hallgren` efter sin "quota". Kolumnen `blocks` visar hur mycket utrymme som redan används (1 block = 1 Kbyte på våra datorer), kolumnen `quota` visar den "mjuka" gränsen för hur mycket man får ha. Man kan tillfälligt överskrida den mjuka gränsen upp till den "hårda" gränsen, som visas i `limit`-kolumnen, men efter en viss tid (som anges av `grace`-kolumnen) kommer då som straff den hårda gränsen att sänkas till precis så mycket man har för tillfällt,

och man kan inte lagra en enda byte till. Man blir då tvungen att ta bort filer tills man kommer under den mjuka gränsen.

Det finns även en gräns för hur många filer man får ha. De fyra sista kolumnerna (*files*, *quota*, *limit*, *grace*) visar hur läget är med antalet filer, men fungerar annars på samma sätt som de fyra föregående kolumnerna.

Hur mycket filutrymme får er labgrupp använda? \_\_\_\_\_

Man borde få en varning (t ex varje gång man loggar in) om har överskridit den mjuka gränsen, men det verkar inte som man får det på E-sektionens datorer. Ta för vana att kolla läget med *quota*-kommandot lite då och då.

### Uppgift 15. Söka information om UNIX-kommandon

Moderna program (som *Netscape*, *NEdit* och *FrameMaker*) har inbyggda hjälpsystem som är rätt vänliga mot användaren. När det gäller information om UNIX-kommandon är den tekniskt fullödlig, men den vänder sig knappast till novisen och man blir lätt skrämmd när man överöses med alla detaljer och finesser. Vet man vad kommandot heter så kan man använda man *kommandonamn* för att få fram ett s k manualblad (ofta flersidigt!). Känner man inte till kommandot kan man använda *apropos ledord*, som listar ett antal kommandonamn - som kan tänkas ha med ledordet att göra - med kort information om varje. Man kan sedan använda man-kommandot för att få ytterligare information. Med *Manual* i programmenyn kan man också leta sig fram till ett kommando och dess manualblad.

Titta på manualbladen för *cd*, *mkdir* och *chmod*.

Använd *apropos* i syfte att hitta t ex redigeringsprogram, pascalkompilatorer, fortran-kompilatorer, skrivarkommandon eller C-kompilatorer. Lämpliga ledord kan vara *editor*, *pascal*, *print*, *fortran* respektive *c* (varför är resultatet av *apropos* i det sista fallet att se som ett misslyckande? Vi återvänder till detta problem i en senare laboration).

Sök reda på tre program som visar vad klockan är (prova programmen när du sökt rätt på dem): \_\_\_\_\_

Det program av de tre ovan vars namn börjar med *xc* visar i normala fall bara en timvisare och en minutvisare. Hur får man programmet att även visa en sekundvisare (använd man-kommandot och lusläs bladet)? \_\_\_\_\_

Ytterligare ett sätt att få fram information om systemet är att använda *Bookreader* som du hittar i Program-menyn. Med den kan man läsa manualer för Digital's systemprodukter, t ex användarmanualer för Pascal- och C-kompilatorerna. Vi struntar i det.

### Uppgift 16. Förgrunds- och bakgrundsjobb

Som bekant kan man ha ett program gående per terminalfönster. Program som startas från bakgrundsmenyn lägger i allmänhet inte beslag på något terminalfönster. I själva verket kan man arbeta med flera program i ett fönster. Om bl a detta handlar denna uppgift.

- Skriv `xclock -update 1` i ett terminalfönster och tryck därefter på return. En klocka startar.
- Stanna programmet `xclock` genom att ge kommandot `<Ctrl-Z>` i terminalfönstret. Observera att du inte avslutat programmet utan bara stannat det.
- Lista det program som du tillfälligt stannat med kommandot `jobs` och starta programmet igen med `fg %1` (eller bara `fg` om `jobs` bara visar detta enda program).
- Stanna programmet igen och starta sedan det i bakgrunden med `bg %1`. Du har nu fått tillbaka redotecknet (promptern) i terminalfönstret trots att programmet `xclock` fortfarande är igång. Du kan nu starta nya program i terminalfönstret.
- Starta `xclock -update 1` en gång till men den här gången skall du starta programmet i bakgrunden direkt. Detta gör man genom att lägga till tecknet `&` sist i kommandot.
- Lägg en av klockorna i förgrunden och avsluta sedan programmet genom att trycka `<Ctrl-C>`. Den andra klockan avslutar vi i nästa uppgift.

### Uppgift 17. Att stoppa skenande program

Ibland händer det att program man startat inte fungerar så bra. Ibland går det kanske t o m så gale t att det normala sättet att avsluta programmet inte fungerar. Man kan då använda kommandot `kill` för att skicka en stoppsignal till programmet. Kommandot används så här:

<code>kill %jobbnummer</code>	Kommandot <code>jobs</code> ger en lista över de program som startats i <i>samma</i> terminalfönster. Denna lista innehåller <i>jobbnummer</i> , som man kan använda om man vill avbryta program.
<code>kill processnummer</code>	Med kommandot <code>ps</code> kan man få en lista över alla program som körs på datorn (egna eller egna+andras). Listan innehåller <i>processnummer</i> .
<code>kill -9 ...</code>	Avslutar ett program med våld. Programmet får ingen chans att städa upp efter sig.

Se även manualbladen för `kill` och `ps`. Använd `man`-kommandot!

- Tag död på klockan från förra uppgiften med `kill`.
- Kör programmet `~hallgren/Intro/mystiskt`. Det kommer att starta en process som du ska avbryta med `kill`. Du kommer att behöva använda `ps` för att ta reda på processnumret. Mer instruktioner skrivs ut då programmet körs.

När du har avbrutit programmet får du ett kvittensmeddelande (olika labgrupper får olika meddelanden). Skriv ner det här:

---

**Anm.** Om man startar ett program som använder för mycket minne blir ofta hela datorn väldig seg och man kanske inte kan göra någonting, inte ens starta ett terminalfönster för att kunna göra `kill` på programmet. Det kan också hända att ett program låser fönstersystemet. Man kan då använda `rlogin` eller `slogin` från ett terminalfönster på en annan dator och köra `kill` på det. Vi återkommer kanske till det i en senare laboration.

### 3 Frivilliga extrauppgifter

#### Uppgift 18. Bilder i e-post

Vi såg tidigare hur man skickar e-post med *Netscape*. Förutom att skriva ett textmeddelande, så kan man också skicka med bilagor, som kan vara godtyckliga filer. På så vis kan man skicka brev med bilder (och ljud etc). Detta förutsätter dock att mottagaren använder *Netscape* (eller liknande) för postläsning, så det skall man nog inte använda okritiskt. För detta används **Attach**-knappen, som låter oss i en dialog ange de filer (GIF m fl), som skall ingå. Prova detta. Skicka brev till den egna gruppen med någon GIF-bild (om inte annat kan du hitta sådana i `~hallgren/Intro/`). Läs brevet.

#### Uppgift 19. Programmering och kompilering

Den här kursen skall inte handla om programmering i något traditionellt programspråk, men kanske kan det ändå vara kul att prova på litet grand.

- Filen `~hallgren/Intro/FirstProg.c` innehåller källkoden till ett korrekt program i programspråket C. Kopiera den till din egen katalog. Kompilera den med kompileringskommandot `gcc` eller `cc`. Kompilatorn lämnar ifrån sig den körbara filen `a.out`. Provkör programmet.
- Programmet beräknar summan av alla heltal från 1 till 1000 genom brutal råräkning. Redigera programmet med *NEdit* så att summan av de 1000000 första positiva heltalen beräknas i stället. Kompilera på nytt och kör det.

# Laborationer i kursmomentet Datoranvändning E1

<http://www.etek.chalmers.se/~hallgren/Eda/>

## Laboration nr 3: Matematikverktyget Maple

1992-1997 Magnus Bondesson  
1998 och 99-11-01 Thomas Hallgren

### 1 Introduktion

Syftet med denna laboration är att ge dig en första praktisk kontakt med matematikverktyget Maple. Vissa av uppgifterna avser dessutom att antyda hur dessa verktyg kan användas för att ge ökad förståelse kring något fenomen. En del uppgifter kanske berör matematiska begrepp som du ännu inte stött på i mattekursen, men det borde gå bra att lösa dem ändå. I en annan laboration kommer vi att bekanta oss med ett annat matematikverktyg MATLAB. Du kommer under andra kurser att möta Maple och MATLAB och lär dig då successivt om systemen efter behov. Man lär sig naturligtvis mycket på att själv experimentera med systemen och prova saker som verkar kul.

#### 1.1 Förberedelser

Innan du kommer till laborationen skall du som vanligt ha tittat på några sidor i Gula Boken, denna gång kapitel 9 om Maple, fram till och med avsnitt 9.7 (hoppa över 9.3). Läs också åtminstone introduktionen i detta laborations-PM.

Där ej annorlunda sägs finns erforderliga funktioner beskrivna i Gula Boken. Så se till att du har den till hands. Labhandledarna svarar naturligtvis också på frågor.

#### 1.2 Redovisning

Visa vad du har gjort för en handledare när du är klar med uppgifterna.

Maple kan ses som en kombination av ett ordbehandlingsprogram och en kalkylator. Medan man arbetar tillverkar man ett dokument där innehållet omväxlande är det man själv har matat in och svar som Maple har räknat ut åt dig. Det är lätt att spara och/eller skriva ut dokumentet. **Spara ofta** under arbetets gång, så att inte allt går förlorat ifall Maple skulle krascha eller det blir strömavbrott.

#### 1.3 Maple i korthet

Maple är ett generellt matematikverktyg som är användbart i alla sammanhang där matematik spelar en roll.

Här följer några grundläggande fakta som kan vara bra att känna till innan man börjar arbeta med Maple:

- Varje kommando skall avslutas med `;` (eller `:` om man inte vill se Maples svar på kommandot).
- I Maple används `:=` för att göra definitioner. Text betyder kommandot `x:=2` att variabeln `x` i fortsättningen står för talet 2.
- Vid redigering flyttar man sig upp i den tidigare textmassan genom att klicka med **vänstra musknappen** eller med **piltangenterna**. Vid utförandet av ett tidigare kommando (eventuellt redigerat) ersätts det tidigare svaret. Med musen eller piltangent kan man återvända till slutet.
- Undvik att trycka på returtangenten om något i textmassan är markerat; det markerade tas nämligen som ett kommando.
- Observera också att liksom i UNIX betyder i Maple stor och liten bokstav olika sak (oftast).

## 2 Uppgifter

### 2.1 Vi bekantar oss med Maple

#### Uppgift 1. Start av Maple

Det kan vara praktiskt att först skapa en katalog (mapp) *maple*, vilket sker med UNIX-kommandot `mkdir maple`, och att med `cd maple` flytta sig till den katalogen. Starta sedan Maple med UNIX-kommandot `xmaple` eller med programmenyns *Maple* (i det sista fallet anser Maple att du är kvar i hemkatalogen). Om Maple-fönstret och arbetsbladet blir små är det idé att omedelbart förstora dem (ändring sparas i filen `.xmaplev4rc`).

#### Uppgift 2. Definitioner, variabler och uttryck

Skriv in följande rader (bortsett från kommentarerna och redotecknen `>`). Varje kommando kommer att resultera i ett svar som ej medtagits här.

```
> s:=2+x;           # Definiera ett uttryck
> x:=1;             # Definiera x
> s;                # Vilket värde har s nu?
> t:=2+x;           # Definiera ett uttryck till
> x:='x';           # Odefiniera x
> s;                # Vilket värde har s nu?
> t;                # Och t?
```

Notera att *s* hela tiden står för uttrycket  $2+x$ , som dock evalueras (beräknas) fullt ut om *x* är definierat, medan *t* — eftersom *x* var definierat när *t* definierades — står för uttrycket 3.

### Uppgift 3. Uttryck

Beräkna  $\pi$  med 100 siffror. Beräkna det därefter även med 200 siffror genom att med musen flytta dig till uttrycket, bara ändra ett tecken i det inmatade och till sist trycka på returtangenten.

### Uppgift 4. Uttryck

Skriv in följande rader (bortsett från kommentarerna och redotecknen >). Varje kommando kommer att resultera i ett svar som ej medtagits här.

```
> 2+1/3; sqrt(99);           # Beräkna värdet av två uttryck
> 2+0.33;sqrt(99.0);        # Två värden till
> 2+Pi; sqrt(99.0+Pi);      # Och ytterligare två
```

Hade vi matat in motsvarande uttryck på en normal miniräknare, hade vi i samtliga fall fått numeriska approximationer. Men här får vi det bara i vissa fall. Vad kan det bero på? Beräkna — utan att skriva om något — numeriska värden i de saknade fallen.

### Uppgift 5. Lär dig redigera och bli imponerad

- Kontrollera att  $x$  är odefinierad genom att skriva ut värdet. Odefiniera, dvs skriv  $x:=x'$ ; om så erfordras.
- Definiera uttrycket  
 $s:=x^8+12*x^7+71*x^6+260*x^5+528*x^4+388*x^3-261*x^2-540*x-459$ ;  
Om det råkar bli fel (lätt hänt när man skriver av sådana här långa konstiga saker), använd då piltangenterna eller musen för att korrigera (se inledningen ovan). I musfallet flyttar du markören till rätt plats, klickar med vänstra musknappen och rättar.
- Skriv ut uttryckets värde.
- Faktorisera uttrycket med *factor*. Resultatet skall om du gjort rätt bli snyggt. Vad blev det? \_\_\_\_\_
- Ändra i uttrycket sista termen till 460 (använd musen för redigeringen) och tryck på returtangenten. Den nya definitionen kopplas till  $s$ .
- Flytta musen till *factor*-raden och tryck på returtangenten. Nu försöker Maple faktorisera det nya uttrycket, men "misslyckas" eftersom uppgiften är "omöjlig". Funktionen *factor* försöker hitta faktorer av lägre gradtal med rationella koefficienter. Speciellt hittas linjära faktorer bara om det finns rationella rötter, d v s av formen  $p/q$ , där  $p$  och  $q$  är heltal.
- Med ovanstående teknik försvinner de tidigare resultaten. Säg att vi vill se både den misslyckade faktoriseringen för 460, som ju syns nu, och den lyckade för 459 utan att behöva skriva om. Då kan vi ta till kopiering/klistring. Markera uttrycksberäkningen, klicka med vänstra mustangenten vid sista redotecknet, klicka med mellersta mustangenten, ändra 460 till 459 och tryck på returtangenten och vips har vi ett nytt uttryck. Alternativt kan du använda **Edit**-menyns **Copy/Paste**. Gör sedan på samma sätt med faktoriseringen.



- h. Med tekniken i g. arbetar vi i två steg. Man kan göra uttrycksberäkningen och faktoriseringen i ett svep genom att skriva de två kommandona efter varandra på samma rad eller på två rader uppdelade med SHIFT-RETURN. Faktorisera på det viset uttrycket med 460 ersatt av 219 genom kopiering/klistring. Hur många faktorer blev det? \_\_\_\_\_

Ett alternativ till klipp-och-klistra-tekniken är att stoppa in en variabel i den delen av uttrycket som vi vill kunna ändra. Se t ex Uppgift 10.

### Uppgift 6. En formel

- a. Tag fram en så enkel formel som möjligt för summan  $\sum_{k=1}^N k^5$ . Använd *factor*. Vad blev formeln? \_\_\_\_\_
- b. Hur många siffror finns det i summans värde för  $N=7000$ ? \_\_\_\_\_  
Tips: använd funktionen *length* (skriv `?length` så får du reda på vad den gör; tag bort hjälpfönstret med vänstra rubrikknappens **Close**).

### Uppgift 7. Grafisk bestämning av nollställe

Ett enkelt och naturligt sätt att bestämma en reell rot till en ekvation  $f(x) = 0$  är att rita upp uttrycket  $f(x)$  på ett allt snävare intervall kring den intressanta roten, vilket ju lätt låter sig göras i Maple (använd `plot(f(x), x=a..b)`; se som vanligt till att  $x$  är odefinierad). Bestäm med denna teknik roten i närheten av  $-2$  till  $x^3 + 2x^2 + x + 1 = 0$  på 0.001 när. Svar: \_\_\_\_\_

### Uppgift 8. Uttryck och funktioner

Ett uttryck definieras enligt modellen `u:=2+3*x`. En motsvarande funktion i stället som `U:=x->2+3*x` (OBS! Skriv inte `U(x):=`, som kan ställa till många underligheter). Vill vi beräkna uttryckets värde för  $x=7$  skriver vi `x:=7; u;`. Det är enklare att beräkna funktionens värde för  $x=7$ . Då skriver vi bara `U(7);`. `U(x)` är naturligtvis ett uttryck. Verktyget *diff* arbetar på uttryck medan *D* bildar derivatan av en funktion (se sid 202 i GB). För att få utskrift av funktionsdefinitionen måste man skriva `print(U);`

Skriv en Maple-funktion motsvarande  $f(x) = x^4 - 1$ . Kalla den t ex FUNK1. Bilda även funktionens derivata och kalla den DFUNK1. Skriv till sist ut  $f(3)$  och  $f'(3)$ .

### Uppgift 9. Numerisk bestämning av rot till ekvation

Givet en ekvation  $f(x)=0$  med en reell rot finns det en käck metod att ta reda på den. Välj först ett tal  $x_0$  i närheten av roten. Upprepa sedan följande beräkning till dess någon form av stabilitet synes uppstå:

$$\text{Ersätt } x_0 \text{ med } x_0 - \frac{f(x_0)}{f'(x_0)}, \text{ där } f'(x) \text{ är derivatan av } f(x).$$

Metoden kallas Newtons iterationsmetod (iterera betyder upprepa) och är lätt att motivera, vilket vi dock inte går in på här.

I Maple gör du lämpligen ungefär så här. Definiera funktioner motsvarande  $f$  och  $f'$ . Sätt  $x_0$  till lämpligt startvärde. Beräkna nytt  $x_0$ -värde enligt formeln. Upprepa det sista steget ett antal gånger genom att bara klicka med musen och trycka på returtangenten.

- Genomför detta för ekvationen  $x^3 + 2x^2 + x + 1 = 0$  och för den rot som ligger i närheten av  $-2$ . Vilket värde stabiliseras beräkningarna vid? \_\_\_\_\_
- Maple har en funktion som heter *fsolve* för numerisk bestämning av rötter. Prova att hitta roten till ekvationen ovan med hjälp av *fsolve*.

### Uppgift 10. Uttryck och olika sorters problem

- Kontrollera att  $x$  och  $a$  är odefinierade. Definiera uttrycket  $s:=x^3+a*x^2+x+1$ ;
- Skriv ut uttryckets värde.
- Sätt  $a$  till 1. Skriv ut uttryckets värde. Faktorisera det med *factor*.
- Sätt  $a$  till 2. Skriv ut uttryckets värde. Försök att faktorisera det med *factor*.
- Sätt  $a$  till  $-37/3$ . Skriv ut uttryckets värde. Faktorisera det med *factor*.
- Sätt  $a$  till 2.1. Skriv ut uttryckets värde. Försök att faktorisera det med *factor*.

Vad lär vi oss av detta? I läroböcker brukar alla problem ha enkla lösningar. Men så är det uppenbarligen inte i verkligheten, inte ens när det gäller faktorisering. Faktoriseringsfunktionen *factor* har de begränsningar som nämndes i Uppgift 5 och som torde sammanfalla med vår uppfattning om faktorisering. Visst skulle man kunna tänka sig att t ex ytterligare några typer av linjära faktorer skulle kunna plockas fram (exempelvis  $x-\sqrt{n}$ ). Sammanfattningsvis: det finns problem som är i sig "omöjliga" och det finns andra problem som det aktuella systemet inte kan klara av.

### Uppgift 11. Integrera

Här följer några uppgifter med bestämda integraler från första delen av ett matematikkompendium (hösten 95), vilka (liksom alla de övriga) lätt löses med Maple. Gör en eller annan och jämför med facit-svaren.

$$80a. \int_0^{\pi/2} \sin^9 x dx \quad 96d. \int_0^{\infty} x^2 e^{-x} dx \quad 300. \int_0^{\pi} \frac{\sin x}{\cos^2 x + 4 \cdot \cos x + 7} dx$$

## 2.2 Grafik

För att rita upp kurvor och ytor har Maple några olika kommandon. Vilket man ska använda beror på vilken form kurvan är given, och om den är tvådimensionell eller tredimensionell:

- Explicit form:**  $y = f(x)$  med  $x \in [a,b]$ , dvs man ger en formel för hur man räknar ut  $y$ -värdet för ett givet  $x$ -värde. (För ytor i 3D blir det  $z = f(x, y)$  med  $x \in [a,b]$ ,  $y \in [c,d]$ .) Kommandon i Maple:

```
plot(f(x), x=a..b);  
plot3d(f(x,y), x=a..b, y=c..d);
```

2. **Implicit form:** kurvan definieras av en ekvation innehållande variablerna  $x$  och  $y$ , t ex  $f(x, y) = 0$ . (För ytor i 3D blir det  $f(x, y, z) = 0$ .) Kommandon i Maple:

```
implicitplot(f(x,y)=0, x=a..b, y=c..d);  
implicitplot3d(f(x,y,z)=0, x=a..b, y=c..d, z=e..f);
```

Kommandona *implicitplot* och *implicitplot3d* ligger inte i standardbiblioteket utan kräver att man först skrivit `with(plots)`;

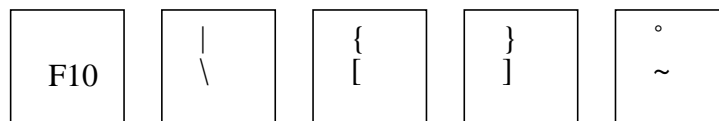
Vill man rita flera kurvor/ytor i samma diagram låter man funktionerna/uttrycken bilda en mängd, t ex:

```
plot({f(x), g(x)}, x=a..b);  
implicitplot3d({f(x,y,z)=0, g(x,y,z)=0}, ...);
```

Genomgående är implicit-rutinerna långsammare än övriga.

### Var finns tecknen [, ], { och }?

Dessa tecken behövs snart. Numera har tangentborden på E alla intressanta tecken (några finns på märkta eller omärkta funktionstangenter till höger om F10 på tangent-



bordet) och när man trycker på motsvarande tangent blir det rätt på skärmen. Precis som det skall vara! Detta under förutsättning att standardinställningarna inte ändrats (det som skall gälla är mellersta bakgrundsmenyens **Keyboard Options/Key Swe Ext**).

### Uppgift 12. Rita i 2D

- Rita kurvan  $y = \frac{\sin(x)}{x}$  på intervallet  $[-10, 10]$  (den är alltså given i explicit form). Prova de olika alternativen i menyn **Axes** (klicka först på figuren så att den är vald) och se efter vad de har för effekt. Förminska/förstora den genom att dra i någon av de små fyrkanterna i markeringsramen.
- Tag med hjälpmenyens **Topic Search** (sökord `color` och sedan **Apply** eller **OK**) reda på hur du kan få kurvan i a) att bli gul. Du kan alternativt använda `?plot` följt av en tryckning på `color` längst ned på den hjälpsida som då visar sig. Rita sedan upp kurvan gulfärgad. Det finns en mängd `s k` optioner till ritkommandona, vissa kan man ändra med menyer.
- Rita upp ellipsen  $x^2+4y^2=1$ ,  $x \in [-1,1]$ ,  $y \in [-1,1]$ . (Här har vi alltså en kurva given på implicit form.) Ellipsen ser ut som en ellips om du i **Projection**-menyn (tryck först på figuren) väljer **Constrained** i stället för det förvalda **Unconstrained**.

### Uppgift 13. Rita flera funktioner samtidigt

Låt  $f(x) = \frac{\sin(x)}{x}$ . Rita upp kurvorna  $y = f(x)$ ,  $y = f'(x)$  och  $y = \int_0^x f(t)dt$  på intervallet  $[-20, 20]$  i en figur.

### Uppgift 14. 3D-ritning

Matematiken bakom detta tas upp först i Matematik del C, så denna uppgifterna får väl ses som en förhandsvisning av vad man kan göra i Maple.

- a. Hitta på en trevlig funktion  $f(x, y)$  av två variabler (t ex

$$f(x, y) = \frac{\sin \sqrt{x^2 + y^2}}{\sqrt{x^2 + y^2}}).$$
 Rita upp den och pröva menyerna **Style**, **Color** och

**Axes** (tryck först på figuren). Tryck på högra musknappen eller R-knappen för att få omritning. Vrid på figuren med vänstra musknappen.

- b. Rita en "igloo" med följande kommando **utan att mata in det** för hand.

```
plot3d([x*sin(x)*cos(y), x*cos(x)*cos(y), x*sin(y)], x=0 .. 2*Pi, y=0 .. Pi);
```

Ge i stället kommandot `?plot3d` och kopiera och klistra in kommandot från exempeldelen av hjälpen (man kallar t för x och u för y men det spelar ju ingen roll)! Vrid på igloon med vänstra musknappen. Se till att ytorna får färg.

- c. Rita upp sfären  $x^2 + y^2 + z^2 = 4$ . Rita sedan upp denna sfär och planet  $x = 1$ . Vrid och vänd.

## 2.3 Dokumentation

### Uppgift 15. Dokumentation

När man skriver rapporter o d använder man normalt något ordbehandlingsprogram. Vi ska i en senare laboration titta på FrameMaker. Man kan foga in figurer från Maple i FrameMaker-dokument. Hur det går till beskrivs i Gula Boken. Men man kan — som omtalas i kompendiet — också göra sin dokumentation direkt i Maple eftersom det går att stoppa in bl a förklarande text i ett arbetsbladet. Det finns åtskilliga möjligheter. Här ser vi bara på en.

Markera en sektionssklammer i arbetsbladets vänsterkant. Välj **Insert/Text Input**.

Skriv in text i den nybildade sektionen. Kontextraden (se figuren på sid 192 i GB) byter samtidigt utseende (sid 211). Pröva eventuellt att välja annan stil än den normala med den vänstra menyn på kontextraden eller använd någon av de följande. Ett exempel ser du till höger.

```
The next example illustrates
• plotting
• colors
> plot(sin(x), x=0 .. 30, color=green);
```

I **Edit**-menyn finns en mängd alternativ för redigering av arbetsbladet, t ex för att ta bort delar, så man kan lätt avlägsna sina misstag före utskrift utan att börja på nytt.

### 3 Frivilliga extrauppgifter

#### Uppgift 16. Integrera

I den övningssexempelsamling i matematik som användes när jag började på Chalmers hittar man bl a integralen  $\int x^n e^{-x} dx$ . Beräkna den för  $n=3$  med Maple. Svaret kan förenklas med *factor()*; En fråga att fundera över: Varför klarar inte Maple integralen för allmänt  $n$ ?

#### Uppgift 17. Integrera

Att beräkna obestämda integraler till rationella funktioner är i de enklaste fallen bara fråga om rutinarbete, eftersom det bygger på (se mattekompndiet) en partialbråksuppdelning av den rationella funktionen. Verkligheten är dock något brutalare, eftersom det inte alltid går att hitta en partialbråksuppdelning av den form man brukar se på i läroböckerna. Men Maple klarar allt (exakt om rationella koefficienter)!

Beräkna med Maple  $\int \frac{x^2}{x^3 + ax^2 + x + 1} dx$  för några olika  $a$ -värden (jfr Uppgift 10). Lämpligen införs först en variabel för uttrycket (se till att  $x$  och  $a$  är odefinierade).

- $a=1$ .
- $a=2$ . I detta fall får man en komplicerad exakt formel som är kopplad till rötterna till en ekvation. Beteckna denna med *integral*. Då ger *evalf(integral)* en numerisk approximation.
- $a=2.1$ . I detta fall får man direkt en numerisk approximation.
- $a=21/10$ . Exakt formel.
- Beräkna ett par av motsvarande bestämda integraler för gränserna 0 och 1.

#### Uppgift 18. Visualisera taylorutveckling

Bestäm polynom av grad 1,3,5,7 och 15 som approximerar funktionen  $\sin(x)$  i närheten av  $x=0$  (se *taylor* och *convert*, sid 201). Lagg för enkelhets skull polynomen i fem olika variabler, t ex  $p1$ ,  $p3$ ,  $p5$ ,  $p7$  och  $p15$ . Rita sedan upp dessa funktioner och  $\sin(x)$  i ett diagram (se sid 176, mitten, eller uppgift 15) för intervallet  $[0,10]$ . Det är lämpligt att i *plot*-kommandot lägga till en sista parameter som anger önskat  $y$ -intervall, t ex  $[-10,10]$ .

#### Uppgift 19. Rita kurvor givna på parameterform

Vi har tidigare sett hur man ritar kurvor givna på explicit och implicit form. En tredje form är **parameterform**, där koordinaterna för punkter på kurvan anges som funktioner av en extra, oberoende variabel, dvs  $x = x(t)$ ,  $y = y(t)$  med  $t \in [a, b]$ . Kommando i Maple:

```
plot([x(t),y(t),t=a..b]);
```

(Iglöom i Uppgift 14 gavs på parameterform.) Rita upp lissajouskurvan  $x = \sin(4t)$ ,  $y = \sin(7t)$  på intervallet  $[0, 2\pi]$ . Prova eventuellt sedan vad som händer om man byter ut konstanterna 4 och 7 mot något annat.

### Uppgift 20. Tillverka sekvenser

Man vill ofta tillverka sekvenser med någon form av regelbundenhet, t ex  $1, 2, 3, \dots, N$  eller  $f(1), f(2), \dots, f(N)$ , där  $N$  är givet och  $f$  en given funktion. Detta kan göras med `seq` (uttryck som innehåller  $i, i=1..N$ ). Som vanligt måste  $i$  (eller det namn vi använder) vara odefinierad eller `'...'`-notation användas (jfr *sum* och *plot*, sid 210).  
T ex

```
> seq(i, i=1..5);  
      1, 2, 3, 4, 5  
> seq(1, i=1..5);  
      1, 1, 1, 1, 1
```

Tillverka de fyra följande sekvenserna (gör det utan att skriva om något i onödan!):

- $1, 8, \dots, N^3$  med  $N=10$
- $\sin(0), \sin(0.01), \dots, \sin(6.28)$
- $[1, 1], [2, 4], \dots, [N, N^3]$  med  $N=10$
- $[0, \sin(0)], [0.01, \sin(0.01)], \dots, [6.28, \sin(6.28)]$

och rita upp punkterna i d. i ett punktdiagram (se GB, sid 206).

### Uppgift 21. Lotto

Vi vill generera 7 slumpmässiga heltal mellan 1 och 35. Tag först med hjälpmenyns **Topic Search** reda på vad `xxxx` nedan skall ersättas med för att följande två rader skall generera två slumpetal i det intervallet (tips: sökord `random+Apply`).

```
> slump:=xxxx(1..35);      # Skapar en funktion för slumpetal  
> slump();slump();        # som vi sedan kan anropa lika många  
                           # gånger som det önskade antalet  
                           # slumpetal
```

Skapa sedan en följd av 7 slumpmässiga heltal (ej nödvändigtvis olika) mellan 1 och 35.

### Uppgift 22. Kobingo med 200 kor och vinster

Vi vill generera ett antal slumpmässiga punkter med heltalskoordinater mellan 1 och 50. T ex om antalet är 2  $(5, 34), (23, 19)$ . Producera med Maple en sekvens av 200 sådana punkter och rita dessutom upp dem i ett punktdiagram (se GB, sid 206). Vad händer om man glömmer `style=POINT` i `plot`-kommandot?

### Uppgift 23. Rotritning

Att rita upp nollställena till ett polynom är lätt i MATLAB, som du kommer att märka senare. I Maple är det omständigare (jag utesluter dock inte att jag missat någon procedur som gör det enkelt). Skriv Maple-kommandon som ritat upp rötterna till  $x^34=1$ . **Ledning:** Vi vet från Gula Boken (sid 205) att *fsolve* ger en sekvens  $z_1, \dots, z_{34}$  av rötterna och att `plot([[x1,y1],...,[xn,yn]], style=POINT)` ritat punkter. Problemet är då att göra om sekvensen till en lista med rötterna på formen  $[x,y]$ . Vi kan bilda en sekvens  $R$  som är just  $[x_1, y_1], \dots, [x_n, y_n]$  med `R:=seq([Re(s[i]), Im(s[i])], i=1..gradtalet)`. Sedan plottar vi bara listan  $[R]$ .

## 4 Maple på webben

Följande webbadresser nämns i Gula Boken:

- <http://www.indiana.edu/~statmath/math/maple/> — diverse Maple-info från Indiana University.
- <http://www.maplesoft.com/> — information om Maple och annat från företaget som säljer Maple.

# Laborationer i kursmomentet Datoranvändning E1

<http://www.etek.chalmers.se/~hallgren/Eda/>

## Laboration nr 4: Dokumentproduktion med FrameMaker

1996: Magnus Bondesson  
Uppdatering 1998 och 99-10-25: Thomas Hallgren

### 1 Introduktion

Denna laboration handlar om dokumentproduktionssystemet FrameMaker, som finns på E-sektionens arbetsstationer. FrameMaker (försvenskat: Ramskaparen) är principiellt sett mycket likt t ex Word för PC och Mac. Idag tar vi upp vad som behövs för att åstadkomma enklare dokument med ett standardiserat utseende: allmänt handhavande; hur man skriver rubriker och vanlig text; hur man skapar bilder, tabeller och matematiska formler. I nästa laboration tittar vi bland annat på hur man kan anpassa och utöka standardmallarna efter eget behov.

#### 1.1 Förberedelser

Innan du kommer till laboration 4 skall du ha läst eller skummat kapitlet om FrameMaker i Gula Boken. Hoppa som vanligt bara över det bland dessa sidor du tycker är konstigt eller för tillfället ointressant och återvänd i stället efter laborationen.

#### 1.2 Redovisning

Skriv ner svaren på frågorna i häftet. Spara det FrameMaker-dokument som tillverkas och visa det för handledaren. Vi kommer även att jobba vidare med dokumentet i nästa laboration.

### 2 Uppgifter

Som tidigare markerar symbolen  i uppgifter med flera delar inledningen på ett stycke med en konkret arbetsuppgift. Dessförinnan kan finnas motiverande eller förklarande text.

#### Uppgift 1. Starta FrameMaker

Starta FrameMaker via bakgrundsmenyn med program. Det finns två alternativ: engelsk eller svensk version. Välj den svenska versionen så att bilderna i Gula Boken stämmer överens med vad som visas på skärmen (bokens bilder är producerade på SUN-datorer med ett något annorlunda fönstersystem, vilket förklarar varför rubrikerna ser litet olika ut på din skärm).



Efter en stund dyker ett litet fönster upp på skärmen med fem knappar. Låt oss titta på dem från höger:



<input type="checkbox"/>	AVSLUTA	Avslutar FrameMaker, så den knappen provar vi inte nu.
<input type="checkbox"/>	INFO	Ger information om version av FrameMaker etc. Prova.
<input type="checkbox"/>	HJÄLP	Öppnar ett fönster där du kan söka information om FrameMaker. Prova att klicka på text, pilar och figurer och se vad som händer. Vi kommer nog oftast att ta oss in i hjälpsystemet på annat sätt.
	ÖPPNA	Använder du då du skall öppna ett befintligt dokument, som du vill redigera eller utgå ifrån. När du klickar på knappen får du upp ett fönster som visar katalogerna (överst) och filerna i din hemkatalog.
	NYTT	Använder du då du skall skapa ett helt nytt dokument och inte har något tidigare att utgå ifrån. När du klickar på knappen får du upp ett fönster där du kan välja bland färdiga mallar.

### Uppgift 2. Titta på ett färdigt dokument

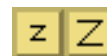
Öppna med **ÖPPNA**-knappen det dokument som du nu håller i din hand. Det finns som `~hallgren/Intro/lab_04.doc`. Vid det här laget har du mött öppningsdialoger i några program. FrameMaker har sin egen variant. Du kan antingen skriva in det fullständiga filnamnet (slutdelen räcker om du är i rätt katalog) överst i dialogen eller klicka dig fram. FrameMaker har inte en ruta för kataloger och en annan för filer, utan i stället räknas först alla kataloger upp och följs av filerna.

Prova nu att (se även Gula Boken, 7.2.3)

a. Flytta dig mellan sidorna med hjälp av hissen och knapparna i fönstrets nederkant. Prova också vad som händer när man flyttar markören över sidgränserna med hjälp av piltangenterna på tangentbordet. Prova även **Föreg-** och **Näst-**tangenterna.



b. Prova förstoringar på olika sätt, med knapparna i fönstrets nederkant.



c. Prova med olika storlekar på fönstret att välja **Anpassa sidan till fönstret**.



d. Prova att ställa in lagom förstoring (t ex 100%) och sedan välja **Anpassa fönstret till sidan**.

e. På sidorna ser du prickade inramningar och diverse styrtecken som inte har med den egentliga texten att göra och inte skrivs ut, men som kan vara till hjälp när man redigerar dokumentet. (Se avsnitt 7.4.2 i Gula Boken.) Lägg speciellt märke till symbolen ¶, som markerar avslutning på ett stycke. Varje stycke har sina egna egenskaper (styckeutformning) och dessa beskrivs med ett styckenamn (styckeutformningsnamn vore korrektare, men tyngre). Flytta markören till stycket efter rubriken för uppgift 3. Styckenamnet kan avläsas i överdelen och underdelen av fönstret. Vad heter stycket? \_\_\_\_\_

Flytta sedan markören till stycket som utgör rubriken för uppgift 3. Styckenamn? \_\_\_\_\_

Dito till laborationstiteln. Styckenamn? \_\_\_\_\_

- f. Visningen av styrtecknen (och även ramar m m) kan om man så önskar stänga av. Använd menyn **Visa/Textsymboler** m fl för detta. Själv tycker jag att det är bra att alltid se dem, men smaken är ju delad. Med dessa hjälpdekorationer avstängda kan man få en bättre uppfattning om hur dokumentet kommer att se ut när man skrivet ut det.

### Uppgift 3. Redigering



Vi skall nu manipulera innehållet i dokumentet en aning.

- a. Flytta markören till en plats i den löpande texten. Skriv in litet text. Pröva **Redigera/Ångra**.
- b. Markera text med dragning. Kopiera med **Redigera**-menyns **Kopiera**, markera ny plats och klistra in med **Redigera**-menyns **Klistra in**. Pröva återigen **Ångra**.
- c. Dubbelklicka i ett ord. Hela ordet skall markeras.
- d. Trippelklicka (dvs tre snabba klick) i ett ord. Hela stycket skall markeras.
- e. Placera markören i ett stycke med namnet *Brödtext* (löpande text brukar i den grafiska branschen kallas så). Tag fram styckekatalogen (med knappen som visas i högra marginalen). Välj *Rubrik 1*. Stycket byter till den valda utformningen. Återgå till det ursprungliga med *Brödtext* eller **Ångra**.
- f. Markera en del av texten i vanlig stil någonstans. Tag fram teckenutformningskatalogen (knappen ser ut som i marginalen). Välj *Tecken fet*. Utformningen ändras. Återgå med *Standardtypsnitt* eller **Ångra**.



Du har nu gjort ett antal förändringar i dokumentet och skulle kunna spara det i den egna katalogen, men det är knappast värt besväret den här gången. Så stäng det bara med **Arkiv/Stäng** utan att spara.

### Uppgift 4. Få mera begrepp om vad FrameMaker kan.

Tryck på **HJÄLP**-knappen. Välj *FrameMaker översikt* och bläddra dig fram till *Exempeldokument*. Titta på dessa: *Rapport*, *Meritförteckning* och *Figurer* genom att dubbelklicka på namnet. Stäng sedan fönstret med **Arkiv/Stäng**. Dessa dokument kan användas som utgångspunkt för egna. *Figurer* i *Figurer* kan markeras, kopieras och klistras in i egna dokument. Vi kan nå dessa exempel enklare via **Hjälp**-menyn i ett dokumentfönster.



### Uppgift 5. Nu gör vi ett eget dokument

Du skall tillverka ett eget dokument utgående från en mall. Det finns många olika mallar, men avsikten är att du skall göra en rapport om litet om det du har upplevt hittills under datoranvändningskursen, så vi väljer en rapportmall. Rapporten skall inte lämnas in eller detaljgranskas, men det begärs att du skriver riktiga meningar och ord.

Däremot får det gärna vara si och så med logiken och sammanhanget. **Dokumentet skall ha den struktur som visas på sista sidan.**

Den här gången lämnar jag inte alltid fullständiga arbetsanvisningar (utan hoppas att du minns litet från föreläsningen eller Gula Boken), så du får slå upp eller fråga en handledare. Tillbringa inte för mycket tid med att leta efter behövlig information på egen hand.

Innan vi sätter igång ett par erinringar: Retur tangenten ger övergång till nytt stycke. ENTER-tangenten (längst ned till höger) ger manuell radbrytning inom ett och samma stycke, men det normala är ju att låta FrameMaker sköta radbrytning automatiskt. Fönster av olika slag som dyker upp och inte har någon egen **Avsluta-** eller **Stäng-**knapp kan stängas genom att man dubbelklickar på knappen längst upp till vänster. Och kanske viktigast: om du märker att du gjort ett misstag, tag det lugnt. Med **Redigera**-menyns **Ångra** kan närmast föregående åtgärd av nästan vilket slag som helst ångras. Men bara närmast föregående, så gör inget förhastat. Det lätt återgå till den senast sparade versionen av dokumentet (med kommandot **Arkiv/Återgå till sparad**) så det kan vara en bra ide att spara dokumentet då och då, när det är i ett bra skick.

- Välj **NYTT**-knappen. Titta eventuellt mycket hastigt på några exempel med färdiga mallar med hjälp av knappen **Standardmallar** i den dialog som visar sig. Text på Brev, Rapport normal och Nyhetsblad. Avsluta varje exempel med **Arkiv/Stäng**.
- Välj **NYTT**-knappen igen. Öppna mallen *Rapport normal*.
- Skriv in rapportens rubrik (styckenamnet *RapportTitel*; avsluta med retur tangenten) och gruppmedlemmarnas namn (styckenamnet *RapportFörfattare*; avsluta med retur tangenten). Som sagt använd vettig text.
- Du har nu fått början av ett stycke, som skall utgöra rubrik (styckenamnet *Rubrik 1*). Fyll i rubriken och tryck på retur tangenten.
- Nu har vi hamnat i början av ett stycke med löpande text (styckenamnet *Text*). Skriv in ett par sådana stycken. Avsluta det sista med retur tangenten.
-  Om vi fortsätter att skriva fortsätter vi med löpande text, men se det vill vi inte, utan vi vill ha en rubrik av samma slag som den tidigare (styckenamnet *Rubrik 1*). Tag därför fram styckekatalogen med översta knappen i figurerna på ömse sidor (svensk till vänster, engelsk till höger). Och välj i katalogfönstret *Rubrik 1*. Notera att styckekatalogen är annorlunda än förut. Tyvärr är mallarnas stycke- och teckenkataloger rätt fattiga. 
- Skriv in ny rubrik och sedan ytterligare ett par stycken med löpande text. Kom ihåg att spara dokumentet då och då (text nu).
- Producera ytterligare två sådana här numrerade grupper. Du kan göra det på samma sätt som i förra momentet, men alternativt genom att du markerar med musen, kopierar med **Redigera/Kopiera**, markerar insättningsplats med musen och klistrar in med **Redigera/Klistra in**.

- Prova **Kontrollera stavning** under **Redigera**-menyn. (Detta kommando skulle säkert kunna ägnas bortåt en timme, men gör inte det.)
- Prova **Synonymordbok** under **Redigera**-menyn.
- Skapa en kopplad ram och rita en bild med FrameMakers ritverktyg (3:e knappen i figurerna ovan) som visar hur det ser ut när du sitter vid en datorskärm. Se avsnitt 7.5 (Rita figurer) i Gula Boken.

Observera att du först måste markera en insättningsplats och skapa en kopplad ram (läs Gula Boken, avsnitt 7.5.1-7.5.3!), för att det du ritar ska få en plats i texten. Saker som ritas utanför en kopplad ram blir liggandes löst på sidan och följer inte med texten.

Notera att verktygsmenyn kan expanderas (avsnitt 7.5.4). Eventuellt kan du hämta någon del av bilden från den i Uppgift 4 nämnda figursamlingen.

Skriv in mer text i ett textstycke ovanför bilden. Kontrollera att bilden flyttas längre ner ihop med den efterföljande texten.

- Hämta nu in en färdig bild (lagrad som `~hallgren/Intro/OS.gif`, i GIF-format), som visar något av enkätresultaten. Placera först markören någonstans i texten. Använd sedan **Arkiv**-menyns **Importerera/Fil**. Välj **Kopiera** till dokumentet och sedan 300 dpi (motsvarande bildstorlek anges till höger). Du kan alltid i efterhand skala om bilden med t ex **Bild**-menyns **Skala**. Bilden ser tyvärr nog inte så bra ut på skärmen men blir OK vid utskrift (om än något liten).
- Vi vill att text skall kunna löpa på ena sidan om bilden ovan. Markera bildramen och välj **Special**-menyns **Kopplad ram**. Ändra i dialogen (se figur sid 137) **Placering** till **Gå in i stycket** och **Justering** till **Vänster** eller **Höger** och tryck på **Ändra ram**.
- Gör sedan en tabell som ser ut ungefär som följer (påhittade värden). Tabeller står det inte något om i Gula Boken, men vi får se om det inte går lätt ändå med **Tabell**-menyns **Sätt in tabell**.

	OS	Antal
	DOS	10
	Windows 95	30
	UNIX	1
	MacOS	7

- Nu vill vi ha in en bild av hur det ser ut på skärmen, t ex en bild av FrameMakers startfönster med de fem knapparna nu inklusive fönsterrubrik. Använd **Arkiv/Hjälpmedel/Avbilda** och sedan **Arkiv/Importerera/Fil** för detta.
- Allra sist skall det naturligtvis i rapporten omtalas att du sysslat med Maple. Och tala om att Maple kan beräkna såväl

$$\sum_{k=1}^N k^5 \text{ som } \int_0^{\pi} \frac{\sin x}{4 \cos x + 7} dx$$

Läs i avsnitt 7.6 om hur du går tillväga för att skapa formler.

### Uppgift 6. Hur skriver man tecken som inte finns på tangentbordet?

Förr eller senare behöver du skriva  $\pi$  i texten eller ordet idé, så det är lika bra att veta hur det går till, eller rättare sagt, hur man tar reda på hur det går till. Använd **Hjälpmenyns Teckenuppsättningar** för detta. Se även avsnitt 7.9.1 i boken.

### Uppgift 7. Skriv ut dokumentet

Ditt dokument borde nu vara tvåsidigt och det tycker jag att vi kan unna oss att skriva ut, så gör det. Men låt först handledaren titta på dokumentet vid skärmen.

Välj **Arkiv/Skriv ut**.

- Notera att man kan välja att skriva ut alla sidor eller bara vissa.
- Det kan vara bra att se till **Sista sidan först** inte är valt. (Detta inställning kan man använda om skrivaren lägger sidorna i "fel" ordningen. Vissa äldre skrivare gör det.)
- Notera att man kan välja vilken skrivare dokumentet ska skrivas ut på. I rutan "Skrivare:" står det antagligen `$PRINTER`, vilket borde ge en skrivare i närheten av den dator du sitter vid. Du kan kontrollera vilken det blir med UNIX-kommandot `echo $PRINTER`). Du kan läsa mer om utskrifter på avsnitt 7.3.8.

Glöm inte att hämta utskriften!

### Uppgift 8. Prova en inbyggd självstudiekurs

FrameMaker har ett inbyggt hjälpsystem, vilket vi redan mött. Det finns också en självstudiekurs (eng. tutorial) uppdelad i ett antal lektioner, som är tänkta att lära användaren olika delar av programmet. Det fina med denna är att man får fin interaktivitet: eleven följer förklaringar och anvisningar och får omedelbar respons på sina åtgärder (både om de är korrekta eller fel). Däremot är det en smaksak om man gillar att läsa sådant här på skärmen eller på papper. Självstudiekursen är skriven helt i FrameMaker och utnyttjar de hypertextmöjligheter och andra finesser som ingår, men vilka vi inte planerar ta upp i denna kurs.



Starta självstudiekursen med **Hjälpmenyns Självstudiekurs** (eller från Hjälpmenyn). FrameMaker vill att alla öppna dokumentfönster stängs först, så låt så ske. Klicka fram listan över vilka lektioner som finns. Titta på vilka avsnitt som ingår i Lektion 1.

När du senare använder FrameMaker i ditt arbete och det är något du inte kan komma på hur man gör, kan det vara bra att komma ihåg att svaret kanske finns i självstudiekursen.

### Uppgift 9. Avsluta

Avsluta FrameMaker med **AVSLUTA**.

### **3 Frivilliga extrauppgifter**

#### **Uppgift 10. Mera självstudier**

Välj någon intressant lektion i självstudiekursen och gå igenom den!

# RAPPORT OM DITT OCH DATT

Magnus Bondesson

## 1.0 Rapportens syfte

Att beskriva ....

## 2.0 Först surfade vi

Det var rätt kul. Men vi upptäckte också att det faktiskt kunde tjäna ett vettigt syfte.

## 3.0 Sedan blev det UNIX

Liknar ju delvis DOS!

## 4.0 Matematik lätt som en plätt

Maple lät oss räkna ut  $\pi$  med 200 decimaler utan vidare.

## 5.0 Och nu ordbehandlas det

Verkar svårt.

# Laborationer i kursmomentet Datoranvändning E1

<http://www.etek.chalmers.se/~hallgren/Eda/>

## Laboration nr 5: Mer om FrameMaker

1996, 1997 Magnus Bondesson  
1998 och 99-10-13 Thomas Hallgren

### 1 Introduktion

I Laboration nr 4 bekantade vi oss med FrameMaker. I denna laboration lär vi oss mer om FrameMaker.

De första övningarna handlar om att skapa egna utformningar och att importera utformningar från ett annat dokument för att helt ändra utseende på ett dokument.

Senare övningar visar hur man skapar numrerade figurer, hur man får stavningskontrollen att fatta vilket språk dokumentet är skrivet på, hur man gör korsreferenser och hur man får en automatiskt genererad innehållsförteckning.

En liten bisak är också att övningarna ger ett exempel på hur man använder kataloger för att få ordning och reda bland sina filer.

#### 1.1 Förberedelser

Som förberedelse inför denna laboration är det lämpligt att läsa de avsnitt i Gula Boken som det finns hänvisningar till i de olika uppgifterna, bland annat avsnitt 7.7 om egna utformningar.

Dokumentet som skapades i Laboration nr 4 återanvänds i denna laboration.

#### 1.2 Redovisning

I övningsuppgifterna står det ibland att man ska spara sitt dokument. Gör det. I slutet är det meningen att du ska ha två FrameMaker-dokument och några andra filer att visa för handledaren. Ett av dokumenten består dessutom av flera filer.

### 2 Uppgifter

Som tidigare markerar symbolen  en konkret arbetsuppgift.

I Laboration nr 4 ägnade vi oss åt FrameMaker och du producerade ett dokument baserat på mallen *Rapport normal*. Nu jobbar vi vidare lite på detta.



### Uppgift 1. Skapa en ny katalog för denna laboration

- Du kommer att tillverka ett antal FrameMaker-dokument och andra filer under denna laboration, så för att hålla ordning, börja med att skapa en ny katalog. Kalla den Lab5. (Du kommer väl ihåg hur man gör? Kolla annars i Laboration nr 2.). Se till att alla filer som skapas under laborationen hamnar i denna katalog!
- Ta fram ditt dokument från Laboration nr 4 och spara det i den nya katalogen Lab5. Kalla det Rapport5a.doc.

### Uppgift 2. Egen styckeutformning

Egna utformningar behandlas i Gula Boken i avsnitt 7.7.

- Det händer då och då att man vill ha en styckeutformning som inte finns färdig i styckekatalogen. Din uppgift här är att skapa en sådan och placera den i styckekatalogen. Den nya utformningen skall heta *Citat*, ha kursiv stil och vara indragen 3 cm i förhållande till vänstermarginalen. Övriga inställningar ska vara samma som för vanlig brödtext. Använd menyn **Utforma/Stycken/Utformning** för att få styckeutformningsfönstret och sedan **Kommandon/Ny utformning** för att få resultatet att hamna i styckekatalogen. (Se 7.7.2 om du vill ha mer detaljerade instruktioner.)
- Prova sedan att utformningen fungerar som den skall.

### Uppgift 3. Egen teckenutformning

- Att göra en ny teckenutformning och lägga till den till teckenkatalogen går till på ungefär samma sätt (se 7.7.3). I teckenkatalogen för ditt dokument finns antagligen *Betoning* som kursiverar markerade tecken. Gör en liknande teckenutformning för nyckelord, och låt dem t ex visas med röd text och fetstil. Övriga egenskaper ska lämnas oförändrade. Kalla den nya teckenstilen *Nyckelord* och lägg den i teckenkatalogen. Använd menyn **Utforma/Tecken/Utformning** för att få teckenutformningsfönstret och sedan **Kommandon/Ny utformning** för att få resultatet att hamna i teckenkatalogen. Det är lämpligt att använda **Kommandon/Ställ in som Låt** vara för att få en teckenutformning som bara påverkar vissa egenskaper.
- Prova att teckenutformningen fungerar. Använd den både på ett ord i brödtexten och ett ord i en rubrik. Kontrollera att orden blir röda och feta och att de behåller sin ursprungliga teckenstorlek.

### Uppgift 4. Spara

- Du har nu ett dokument med en egen styckeutformning och en egen teckenutformning och några exempel som visar att det fungerar. Spara dokumentet. (Filen ska fortfarande heta Rapport5a.doc). Se till att det hamnar i katalogen Lab5. Ändringarna i nästa uppgift ska sparas i en annan fil.

### Uppgift 5. Ändra stil på hela dokumentet

Om man har utformat sitt dokument genom att konsekvent använda de namngivna utformningarna i stycke- och teckenutformningskatalogerna kan man ganska lätt

ändra stil på hela dokumentet genom att importera utformningar från ett annat dokument.

Tillvägagångssätt:

- Förutom det dokument man arbetar med måste man öppna det dokument man vill importera utformningar från.
- Sedan väljer man **Arkiv/Importera/Utformningar...** från menyn. I fönstret som visas finns en meny där man kan välja vilket dokument man vill importera från. Här visas andra dokument som är öppna. (Det är därför man måste alltså öppna det dokument man vill importera utformningar från först.) Nyskapade dokument visas inte förrän efter man har sparat dem första gången.

Det finns också en mängd knappar för att ställa in vad som ska importeras. Alla alternativ är påslagna från början och det är väl lämpligt om man vill man ändra stil helt och hållet på sitt dokument.

Prova att importera utformningarna från standardmallen `RapportNumerisk` eller `RapportKantrubruk` i ditt eget dokument. Du måste alltså skapa ett nytt dokument från den valda mallen och spara det för att kunna importera utformningar från det. (För att se finessen med `RapportKantrubruk` bör man ha några stycken med utformningen *Rubrik2* i sitt dokument.)

Spara det omformade dokumentet under ett nytt namn, `Rapport5b.doc`. Detta dokument arbetar vi inte vidare med, utan vi fortsätter nu med det första dokumentet, `Rapport5a.doc`.

### Uppgift 6. Numrerade figurer

I rapporter är det vanligt att man numrerar figurer och hänvisar till dem från ett eller flera ställen i den löpande texten.

Det finns inget färdigt verktyg för att skapa numrerade figurer i FrameMaker. I dokumentmallen *Rapport Normal* finns det en styckeutformning som heter *Bild*, som det är tänkt att man ska använda när man skapar en numrerad figur. Själva figuren gör man plats för genom att sätta in en kopplad ram i stycket (detta övade vi på i Laboration nr 4). Ramens placering ska vara **Under aktuell rad**.

Öppna ditt första dokument (`Rapport5a.doc`) och prova att göra en figur på ovan beskrivna sätt.

Detta sätt att göra figurer är enkelt men det har vissa begränsningar. Gör även Uppgift 13 om du vill lära dig ett annat sätt, som ger mer flexibilitet i placeringen av figuren.

Spara dokumentet (`Rapport5a.doc`).

### Uppgift 7. Språkinställningar och stavningskontroll

(Stavningskontrollen beskrivs endast kortfattat i Gula Boken, avsnitt 7.9.2. Språkinställningar behandlas inte alls.)

FrameMakers stavningskontroll kan kontrollera dokument skrivna på många olika språk, men hur vet FrameMaker på vilket språk dokumentet är skrivet? Man skulle kunna tro att man får engelsk stavningskontroll om man kör den engelska versionen av FrameMaker, och svensk stavningskontroll då man kör den svenska versionen. Men det är faktiskt lite finurligare gjort än så.

Som vi sett tidigare finns det en stor mängd inställningar i styckeutformningar. Här finns även en språkinställning, som man får fram genom att välja **Egenskapar: Special** i styckeutformningsfönstret. Det är denna inställning som används vid stavningskontroll. (Den används även för synonymordboken och avstavning.) Om man utgår från ett svenskt malldokument kan man räkna med att styckeutformningarnas språk är svenska.

Genom att göra lämpliga styckeutformningar kan man alltså ha flera språk i dokument utan att stavningskontrollen klagat. På Chalmers är det t ex vanligt att rapporter har både en svensk och en engelsk titel, och en engelsk sammanfattning även om de är skrivna på svenska.

Lägg till ett stycke skrivet på engelska<sup>1</sup>. Skapa en ny styckeutformning för brödtext med engelska som språk och använd den. Stavningskontrollera dokumentet och notera att både den svenska och den engelska texten godkänns.

Spara!

### Uppgift 8. Mallsidor

Läs i Gula Boken (avsnitt 7.8) om mallsidor.

Tag fram mallsidorna för ert dokument. Det borde vara baserat på mallen *Rapport Normal*, i vilken det finns sidnummer i sidfoten. Lägg till sidnummer även i sidhuvudet. Lägg även till ert labgrupsnummer och era namn i sidhuvudet.

Spara!

### Uppgift 9. Hänvisningar (korsreferenser)

Läs i Gula Boken, avsnitt 7.9.7, om hur man gör korsreferenser.

I Uppgift 6 gjorde du en numrerad figur. Skriv lite ny text innehållande en korsreferens till den figuren. Välj en passande korsreferensutformning.

Spara!

### Uppgift 10. Böcker – dokument som består av flera filer

(Detta beskrivs inte i Gula Boken. En del information kan man hitta i den inbyggda hjälpen i FrameMaker. Välj **Innehåll...** i **Hjälp**-menyn och klicka på *Generera filer*, så visas längst till höger i fönstret en lista med relevanta ämnen.)

---

1. Jag antar här att resten av dokumentet är skrivet på svenska. Skulle det vara skrivet på engelska, skriv det nya stycket på något annat språk, t ex svenska.

FrameMaker tillåter att dokument är uppdelade i flera filer. Detta gör det möjligt för flera personer att samtidigt arbeta på var sin del av dokumentet. Man kan sedan skapa en så kallad *bok* där man räknar upp vilka filer som ingår i dokumentet. En bok kan även innehålla automatiskt genererade filer, t ex innehållsförteckningar och index.



Skapa en bok genom att använda kommandot **Arkiv/Generera/bok**, och i fönstret som dyker upp välja alternativet **Ny Bok**, och trycka på knappen **Generera**. Du har nu fått en ny bok som innehåller en fil. Spara den under namnet `Rapport5a.Bok` (vilket borde vara det namn FrameMaker föreslår).

Några fakta om böcker som kan vara bra att känna till innan vi fortsätter:

- Man kan fortsätta att arbeta som vanligt på de enskilda filerna som tillhör en bok, men i bokfönstret kan man även utföra kommandon som arbetar på hela dokumentet på en gång. Med bokfönstrets **Arkiv/Skriv ut** kan man t ex skriva ut hela dokumentet på en gång.
- Man kan öppna filerna som ingår i en bok genom att dubbelklicka på dem i bokfönstret.
- Det mesta som fungerar automatiskt när hela dokumentet finns i en fil fortsätter att göra det även när dokumentet är uppdelat i flera filer. FrameMaker håller t ex reda på sidnummer och styckennummer så att numreringen i nästa fil fortsätter där det föregående slutar. Man kan även göra korsreferenser till andra filer. Man måste dock använda ett speciellt kommando, **Arkiv/Generera/uppdatera** i bokfönstret, för att FrameMaker ska gå igenom alla filerna och göra de eventuella justeringar som behövs när något dokument har ändrats. FrameMaker behöver alltså öppna alla filerna, så man kan inte göra detta om någon annan samtidigt redigerar någon av filerna. (Det gör inget om man själv har filer öppna, men de kan behöva sparas efteråt.)

### Uppgift 11. Automatisk generering av innehållsförteckning

(Detta beskrivs inte heller i Gula Boken)



Vi ska nu lägga till en innehållsförteckning till den ovan skapade bokfilen.

- Välj kommandot **Lägg till fil...** i bokfönstrets **Arkiv**-meny. I fönstret som dyker upp kan man välja mellan att lägga till en automatiskt genererad fil av något slag, eller en vanlig fil.
- Välj alternativet **Genererad lista** och från menyn intill **Innehållsförteckning**. (Antagligen är dessa alternativ redan valda när fönstret öppnas.)
- Välj på vilken plats i dokumentet innehållsförteckning ska placeras. Detta görs i listan till höger i fönstret. Man väljer en fil i listan och sedan **Lägg till före** eller **Lägg till efter** i menyn ovanför listan. I detta fall finns det väl bara en att välja på, och innehållsförteckningen bör väl placeras först i dokumentet.
- Tryck på knappen **Lägg till**.

- Nu visas ett nytt fönster. Här ska man ställa in vad man vill ha med i innehållsförteckningen. Detta gör man genom att välja vilka styckeutformningar som ska tas med. Ta med de utformningar som används för rubriker!
- Antagligen visas fönstret för att lägga till filer igen, men vi har inga mer tillägg att göra nu, så tryck på knappen **Klar**.
- Spara bokfilen!

I bokfönstret ser vi nu två filer: innehållsförteckningen (`Rapport5aIH.doc`) och den ursprungliga filen (`Rapport5a.doc`). Efter innehållsförteckningens namn visas ett **+**-tecken, som betyder att filen genereras automatiskt.

**Anm.** Man kan ändra ordningen på dokumenten i efterhand med kommandot **Arkiv/Ordna om filer...**

- Ännu finns det ingen innehållsförteckning. Den skapas först när man använder det ovan nämnda kommandot **Arkiv/Generera/uppdatera** i bokfönstret. Gör det nu! Ett nytt dokumentfönster med en innehållsförteckning öppnas. Spara det!

Du kanske inte är helt nöjd med innehållsförteckningens utseende. Gör i så fall även Uppgift 14.

### Uppgift 12. Generera PostScript-filer

Hur gör man om man vill göra sina FrameMaker-producerade dokument tillgängliga för andra via webben? Ett sätt är förstås att kopiera FrameMaker-filen till sin webbplats, men då måste den som vill titta på dokumentet ha FrameMaker, och det är det ju långt ifrån alla som har. Ett bättre alternativ är att skapa en PostScript-fil.

PostScript är det format FrameMaker använder för utskrift på skrivare, så att skapa en PostScript-fil är en variant på att göra en utskrift. Man väljer alltså **Skriv ut...** i **Arkiv**-menyn och kryssar i rutan **Skriv till fil**. FrameMaker föreslår ett filnamn, men man kan ändra på det om man vill.

- Skapa en PostScript-fil för hela ditt dokument (`Rapport5a.Bok`). Kalla filen `Rapport5a.ps` (vilket antagligen är vad FrameMaker föreslår).
- Med kommandot `ghostview filnamn` i ett terminalfönster kan man titta på en PostScript-fil. Ta en titt på filen du just skapade!

Med kommandot `lpr filnamn` kan man skriva ut PostScript-filer (och enkla textfiler). Om du vill kan du prova att skriva ut ditt dokument på detta sättet.

## 3 Frivilliga extrauppgifter

### Uppgift 13. Ett annat sätt att göra numrerade figurer

En vanlig stil när det gäller placering av figurer är att de läggs under den rad där man refererar till figuren, om det finns plats. Annars låter man den löpande texten fortsätta och lägger figuren överst på nästa sida. Denna metod får man om man kryssar för alternativet **Flytande** när man skapar (eller ändrar) kopplade ramar i FrameMaker.

Nackdelen med den metod för numrering av figurer som beskrevs i Uppgift 6 är att bildtexten inte följer med figuren om figuren hamnar på något annat ställe än under aktuell rad. Prova t ex att markera ramen du nyss gjorde och välj placeringen **Överst i spalten**.

Ett annat sätt numrera figurer är att använda tabeller istället för kopplade ramar. Dokumentet `~hallgren/Intro/Figurer.doc` innehåller ett exempel och mer instruktioner. Öppna det och ta en titt.

Importera stycke- och tabellutformningarna från `Figurer.doc` i ditt eget dokument och skapa en numrerad figur.

### Uppgift 14. Formatering av innehållsförteckningar

Det finns ett flertal sätt att påverka innehållsförteckningens utformning:

- När man skapar innehållsförteckningen, och i efterhand genom att använda kommandot **Arkiv/Ställ in filalternativ...** i bokfönstret, kan man välja vilka rubriker som ska komma med i innehållsförteckningen. Effekten av denna ändring syns inte förrän efter man har gjort **Arkiv/Generera/uppdatera** igen.
- FrameMaker skapar nya styckeutformningar för innehållsförteckningen. Om man t ex har valt att ta med stycken med utformningen *Rubrik 1* i innehållsförteckningen skapar FrameMaker styckeutformningen *Rubrik 1IH* för motsvarande poster i innehållsförteckningen. I dessa styckeutformningar kan man ändra t ex typsnitt, teckenstorlek, indragning och avstånd ovanför/nedanför stycket så att innehållsförteckningen blir lättläst och får ett utseende som passar ihop med resten av dokumentet.
- Man kan lägga till egen text i innehållsförteckningen. Denna bevaras när man gör **Arkiv/Generera/uppdatera**, om den har en vanlig styckeutformning, dvs inte någon av de nya styckeutformningarna som slutar på *IH*.
- Förutom text- och mallsidor innehåller FrameMaker-dokument även s k referenssidor som man kan titta på genom att välja **Visa/Referenssidor** i menyn. I innehållsförteckningsfilen finns det en referenssida med en mall (ett så kallat *specialflöde*) som bestämmer vilken text varje post i innehållsförteckningen ska innehålla. För varje styckenamn som man har valt att ta med i innehållsförteckningen finns en rad i mallen som talar om hur stycken med den utformningen ska presenteras i innehållsförteckningen.

Raderna i mallen kan innehålla omväxlande vanlig text och variabler. Variablerna fungerar på samma sätt som variabler i korsreferensutformningar, och kommer att ersättas med text ur dokumentet då innehållsförteckningen genereras. Exempel på användbara variabler är `<$paratext>`, `<$paranum>` och `<$pagenum>`. Dessa ersätts med texten i stycket (t ex själva rubriken), styckets nummer (t ex en ett kapitelnummer) respektive numret på sidan där stycket finns.

Mer detaljerade instruktioner och exempel kan man hitta i FrameMakers hjälpfiler. Använd, som ovan, kommandot **Innehåll...** i **Hjälp**-menyn. Klicka på **Index** leta upp *innehållsförteckningar*, *specialflöden för i indexet*.

Effekten av ändringar i mallen syns inte förrän efter man har gjort **Arkiv/Generera/uppdatera** igen.

- Snygga till innehållsförteckningen i ditt dokument. Från början ser den antagligen ut ungefär som till vänster nedan. Gör de förändringar som krävs för att den ska se ut ungefär som till höger.

1.0	Rapportens syfte <sup>3</sup>
2.0	Först surfade vi <sup>3</sup>
2.1	Vi skickade post och läste nyheter <sup>3</sup>
3.0	Sedan blev det UNIX <sup>3</sup>

<b>1.0</b>	<b>Rapportens syfte.....</b>	<b>3</b>
<b>2.0</b>	<b>Först surfade vi .....</b>	<b>3</b>
2.1	Vi skickade post och läste nyheter.....	3
<b>3.0</b>	<b>Sedan blev det UNIX .....</b>	<b>3</b>

Tips: det enda som krävs är några ändringar i styckeutformningarna. Att sidnumren inte blir högerjusterade från början (fast det kanske fungerar när ni gör det) beror bara på att ett tabstopp ligger för långt till höger och därför ignoreras.

Fråga handledaren om hjälp om det verkar för krångligt!

## 4 Avslutande kommentarer

Nyare versioner av FrameMaker (t ex 5.5) kan generera både HTML och PDF (Portable Document Format). Det senare är en variant på PostScript som har blivit populär för elektronisk spridning av dokument. För att titta på PDF-filer kan man använda *Acrobat Reader*, som är gratis och finns för flera olika datorsystem.

Den version av FrameMaker som finns på E:s datorer är 5.1, så tyvärr kan vi inte öva på att skapa PDF-filer. Gula Boken, som du tittade på i Uppgift 9 i Laboration nr 1, är ett exempel på en PDF-fil genererad med FrameMaker.

FrameMaker kan läsa dokument skapade av vissa andra ordbehandlingsprogram och även spara dokument i andra filformat. Vill man flytta filer mellan FrameMaker och Microsoft Word blir det antagligen bäst om man sparar dokumentet i formatet Microsoft RTF (Rich Text Format). Både Word och FrameMaker kan läsa och skriva detta format. Dock kan man inte räkna med att all formatering och alla inställningar bevaras exakt, så man behöver antagligen göra vissa justeringar manuellt efter konverteringen.

### 4.1 Information på webben

Det är företaget Adobe som gör både FrameMaker och Acrobat Reader. Adobe har webbsidor på [www.adobe.se](http://www.adobe.se) och [www.adobe.com](http://www.adobe.com). Man kan hitta information om de två programmen på t ex sidorna

<http://www.adobe.se/produktinfo/framemaker/>

<http://www.adobe.com/prodindex/acrobat/>

På föreläsningen gick vi bland annat igenom utskrifter i Unix och vanliga misstag vid ordbehandling. Sammanfattningar av detta finns nu på webben. Se sidan

<http://www.etek.chalmers.se/~hallgren/Eda/förel.html>

# Laborationer i kursmomentet Datoranvändning E1

<http://www.etek.chalmers.se/~hallgren/Eda/>

## Laboration nr 6: Mer om UNIX

1996, 1997 Magnus Bondesson  
1998 och 99-11-21 Thomas Hallgren

### 1 Introduktion

Vi stötte redan i början av läsperiod 1 på några de vanligaste UNIX-kommandona. I denna laboration presenteras några fler användbara kommandon, tillsammans med sätt att kombinera kommandon för att utföra uppgifter som det inte finns något färdigt kommando för.

Kommandon som dyker upp är: `finger`, `rwho`, `v`, `w`, `uptime`, `xload`, `which`, `echo`, `set`, `wc`, `grep`, `sort`, `cut`, `chmod`.

Begrepp som dyker upp: variabler, omdirigering, rör (pipes), kommandofiler.

#### 1.1 Förberedelser

Gå på föreläsningen! Skumma igenom häftet och läs de avsnitt i Gula Boken som det finns hänvisningar till!

#### 1.2 Redovisning

Skriv svar på frågorna i häftet. Spara filer som skapas under arbetet. Visa allt för handledaren för att bli godkänd.

### 2 Uppgifter

Som tidigare markerar symbolen  i uppgifter med flera delar inledningen på ett stycke med en konkret arbetsuppgift. Dessförinnan kan finnas motiverande eller förklarande text.

#### 2.1 Diverse användbara UNIX-kommandon

##### Uppgift 1. Vem använder datorerna? Finns det någon ledig dator?

Pröva följande kommandon. Se manualblad eller Gula Boken för mer information.

- `finger användarnamn` eller `förnamn` eller `efternamn`  
Ger information om alla användare med aktuellt namn.



- `finger användarnamn@datornamn`.  
Ger information om en användare på en annan dator. Detta brukar fungera om den andra datorn är en Unix-dator och `finger`-servern inte är avstängd av säkerhetsskäl. Prova `tex finger hallgren@butthead.cs.chalmers.se`.
- `rwho`  
Ger information om alla påloggade användare vid E-systemet.
- `w`  
Visar bland annat vilka program användare på samma dator kör för tillfället.
- `v`  
Lokalt UNIX-kommando, som ger information om lediga/upptagna datorer. `v -h` ger mer info om vad kommandot kan göra.
- `uptime`  
Visar medelvärden över hur hårt belastad datorn har varit den senaste tiden: 0.00 = datorn har inget att göra, 0.50 = datorn arbetar med något 50% av tiden, 1.00 = datorn är upptagen med något hela tiden, 2.00 = datorn har i genomsnitt två olika saker att göra och växlar mellan dem, o s v. Programmet `xload` (som normalt startas automatiskt och visas i övre högra hörnet av skärmen) visar samma sak i ett diagram.

## Uppgift 2. Var finns alla kommandon?

Kommandon i UNIX är program, som finns i filer. Med kommandot `which` kan man ta reda på var ett kommando finns. Kommandot `which ls` ger svaret `/usr/bin/ls`, t ex.

- a. Var finns kommandot `date`? \_\_\_\_\_
- b. Var finns kommandot `v`? \_\_\_\_\_
- c. Var finns `netscape`? \_\_\_\_\_

Hur kommer det sig att det räcker att skriva `ls` för att kommandotolken ska hitta `/usr/bin/ls`? Hittar kommandotolken rätt program var det än finns? Det rätta svaret är att kommandotolken letar i de kataloger som anges av variabeln `$path`. Man kan använda kommandot `echo $path` för att ta reda på hur den är definierad.

- d. Hur många kataloger letar kommandotolken i? \_\_\_\_\_

Man kan lätt själv ändra vilka kataloger kommandotolken ska leta i. Antag t ex att vi vill lägga till `~hallgren/Intro/` i listan. Vi kan då skriva

```
set path = ( $path ~hallgren/Intro/ )
```

vilket betyder att det nya värdet på variabeln blir det gamla utökat med `~hallgren/Intro/`.

**Anm.** Om det finns två kommandon med samma namn får man det som finns i den katalog som står först i listan.

**Anm.** I hemkatalogen finns en dold kommandofil som heter `.cshrc`. Kommandona i den utförs varje gång en ny kommandotolk startas (t ex när man startar ett terminal-

fönster). Man kan stoppa in egna kommandon här, t ex för att lägga till egna kommandokataloger till `$path`-variablen.

**Anm.** `$path` är bara en av många variabler som styr hur kommandotolken fungerar. Se avsnitt 1.16.4 Gula Boken för mer information. Det finns även en uppsättning variabler som påverkar andra kommandon, t ex `$PRINTER` för val av skrivare, som vi stötte på i Laboration nr 4. Dessa beskrivs i avsnitt 1.16.5.

## 2.2 UNIX-kommandon, omdirigering och rörledning

### Uppgift 3. Omdirigering

Normalt hamnar resultaten av ett UNIX-kommando i det fönster i vilket kommandot givits, vilket brukar kallas "standard-ut". Men man kan också dirigera resultaten så att de i stället hamnar i en fil, med (`>` kan utläsas "från kommandot till")

```
kommando >filnamn
```

Läs mera om omdirigering i avsnitt 1.16.1 i Gula Boken.



Se till att resultatet av `ls -l` hamnar som filen `LISTA`. Kontrollera med t ex `more` att `LISTA` innehåller det den skall.

På motsvarande sätt kan ett kommando som normalt hämtar information från tangentbordet, "standard-in", ta den från en fil. Man använder då i stället tecknet `<` ("till kommandot från"). Vi avstår från någon uppgift på detta.

### Uppgift 4. Några användbara UNIX-kommandon: textfiler

Under en stor del av UNIX' livstid (hela 70-talet och större delen av 80-talet) var det dominerande sättet att arbeta med datorn via text-terminaler. I de flesta UNIX-system finns det därför fortfarande en uppsjö av program som bearbetar textfiler på olika sätt. Det skulle vara ganska jobbigt att lära sig hur alla fungerar, men eftersom deras funktionalitet överlappar en hel del behöver man inte det heller. Vi kommer nedan att stöta på några av de vanligaste, som kan vara bra att känna till för husbehov.

Låt oss till att börja med titta på tre kommandon `wc`, `grep` och `sort` som alla kan användas på två olika sätt. Om man skriver ett eller flera filnamn på kommandoraden läser programmen indata från dessa filer. Annars tas indata från "standard-in", dvs det man skriver på tangentbordet på raderna efter kommandot, om man inte har gjort någon omdirigering med "`<`". Vid inmatning på detta sätt anges slut på indata med CTRL-D. Mer information om kommandona fås på vanligt sätt (manualblad resp sid 58).



Tag med hjälp av `wc` reda på antal rader, ord och tecken i filen `Berling.txt`. Svar:



Kommandot `grep` används för att söka efter teckenföljder i en fil. När `grep` funnit en rad som innehåller den sökta textsträngen så skriver `grep` ut denna rad. Skriv ut de rader i filen `Berling.txt` som innehåller teckenföljden `hon`. Ta också reda på vilka som innehåller *ordet* `hon`. (Tips: läs vad `grep -w` betyder i manualbladet för `grep`.)

- Kommandot `sort` används för att sortera text eller tal. Sortera raderna i `Berling.txt` alfabetiskt.

### Uppgift 5. Rörledningar (pipes)

I denna uppgift ska vi göra några övningar med rörledningar, beskrivs kortfattat i Gula Boken, sid 23. Genom att känna till några enkla grundkommandon och hur man kopplar samman kommandon med rör, kan man med ganska lite arbete lösa till synes ganska komplicerade uppgifter. Vi avslutar denna uppgift med att konstruera en alfabetisk lista med namnen på de som nu går i E1.

UNIX är ett fleranvändarsystem. Detta betyder bland annat att man måste logga in innan man kan börja använda datorn, att alla filer har en ägare som bestämmer vilka andra användare som ska få läsa och ändra i filen. Någonstans i systemet måste det alltså finnas information om vilka användare som har rätt att använda datorn. I UNIX finns denna information traditionellt i filen `/etc/passwd`. I filen finns för varje användare bland annat användarnamn, lösenord, hemkatalog och vald kommandotolk. En trevlig egenskap hos denna fil är att den är en vanlig textfil och att alla användare har rätt att titta i den.

- Tag en titt på filen `/etc/passwd` med `more` (eller `less`)!

Som kanske framgår beskriver varje rad i filen en användare. Raderna består av ett antal fält som är åtskiljda av kolon. De första tvåfälten innehåller användarnamnet och lösenordet<sup>1</sup>. Det femte fältet är användarens riktiga namn.

**Anm.** Mer information kan fås med kommandot `man 4 passwd`, där 4 ger avsnitt 4 i manualen, som beskriver olika filformat. (Om man inte anger något avsnitt får man 1, som beskriver kommandon.)

- Hur många registrerade användare finns det i Etek-systemet? \_\_\_\_\_

- De som är inskrivna på E i år har sin hemkatalog i `/u1/e99`. Hur många är inskrivna i år? (Tips använd kommandona `grep` och `wc`, och ett rör) \_\_\_\_\_

Kommandot för att räkna ut detta blev: \_\_\_\_\_

Slutligen ska vi konstruera namnlistan och sortera den. Man kan använda kommandot `cut` för att välja ut vissa fält ur varje rad i en fil. I vårt fall blir det `cut -f 5 -d :` (eftersom vi vill välja ut det femte fältet ochfälten skiljs åt av kolon. Se man `cut` för förklaring). För sorteringen antar vi lite förenklat att efternamnet är det andra namnet. Sorteringen kan då göras med `sort +1`, där `+1` betyder att `sort` ska bortse från första ordet på varje rad vid sorteringen.

- Sätt nu samman alltihop till ett kommando som sparar namnlistan i filen `E1-namnlista`. Hur blir kommandot?

- Utför kommandot och spara filen!

<sup>1</sup> Lösenorden lagras förstås inte i klartext, utan är krypterade...

### Uppgift 6. En nyttig användning av rör

I Uppgift 15 i Laboration nr 6 provade vi att söka information om kommandon genom att använda `apropos`. Ett problem är att `apropos` hittar väldigt många svar om det man söker efter har ett kort namn. Vill man hitta information om C-kompilatorer får man t ex inte mycket hjälp av `apropos c` (prova!). En lösning på detta är att kombinera `apropos` med `grep`.

För att hitta information om C-kompilatorer kan det vara lämpligt att söka efter `compiler` med `apropos` använda `grep` för att välja ut de rader i svaret som innehåller ordet C. Hur blir kommandot för detta?

---

Hur många rader långt blir svaret då man utför kommandot? (Tips: räkna raderna manuellt, eller kombinera kommandot med `wc`.) \_\_\_\_\_

## 2.3 Kommandofiler

### Uppgift 7. Kommandofiler

När man startar ett terminalfönster startas normal kommandotolken `tcsh`, som läser de kommandon man matar in på tangentbordet och utför dem. En naturlig fråga är nu: kan `tcsh`, liksom kommandona vi tittade på i Uppgift 4, även läsa indata från en fil? Svaret är ja! Har man en fil som innehåller kommandon kan man få `tcsh` att utföra dem genom att skriva `tcsh filnamn`.

Filer som innehåller kommandon kallas inte helt oväntat för **kommandofiler** (se sidan 29 i Gula Boken). En kommandofil är alltså en textfil (tillverkas med t ex *NEdit*) bestående av ett antal kommandorader.

Gör med *NEdit* en fil med innehållet

```
date
echo $1 $USER
```

Spara den under namnet `kommando1`. Provkör kommandofilen genom att mata in t ex

```
tcsh kommando1 Hejsan
```

Vad skrivs ut? \_\_\_\_\_

Förklaring: `$1` är en variabel som ersätts med det ord som skrivs efter `kommando1` och `$USER` ersätts med användarnamnet. Kommandot `echo` skriver ut en textföljd, dvs kommandofilanropet kommer att skriva ut datumuppgifter och sedan Hejsan *användarnamnet*.

Det går även att göra kommandofiler lika lätta att använda som vanliga kommandon (dvs man slipper skriva namnet på kommandotolk varje gång man använder dem). `kommando1` görs körbar med UNIX-kommandot `chmod` enligt

```
chmod +x kommando1.
```

Därefter används kommandofilen med t ex

```
kommandol Hejsan
```

Prova detta!

**Anm.** Kommandofiler som används på detta sätt tolkas av kommandotolken `sh`, inte `tcsh`, som på E är den normala kommandotolken för det man själv skriver in i ett terminalfönster. I grunden fungerar båda på samma sätt, men lite mer avancerade saker skrivs på olika sätt. Vill man ange explicit vilken kommandotolk en kommandofil ska tolkas av kan man ange detta på första raden i filen. För att få `tcsh` kan man skriva `#!/bin/tcsh -f`

### Uppgift 8. Kommandot `e9rwho`

Tillverka ett nytt kommando som fungerar som `rwho` (som vi provade i Uppgift 1), men utskriften begränsas till användare vars namn börjar på `e9`. Kalla kommandot `e9rwho`.

### Uppgift 9. Finns det några handledare i närheten?

Om man inte ser någon handledare när man vill bli godkänd kan man ju alltid kolla om någon handledare är inloggad. Kommandot `~hallgren/Intro/vvh` använder `v` (se Uppgift 1) och en variant på `grep` för att konsturera en lista med inloggade handledare. Prova det! Titta också hur det är gjort. Lägg märke till att kommandofiler kan innehålla kommentarer (text som kommandotolken struntar i) som föregås av `#`. Vad heter varianten av `grep` som används? \_\_\_\_\_

### Uppgift 10. Kommandot `inskrivna`

Tillverka ett nytt kommando med namnet `inskrivna`, som ger en lista med namnen på de som är inskrivna på E ett visst år. Med

```
inskrivna 99
```

ska man alltså få namnlistan vi tillverkade i Uppgift 5, men kommandot ska även fungera för andra årtal. Hur många av de som skrevs in 1989 har fortfarande kvar sitt konto på Etek-systemet? \_\_\_\_\_

## 3 Avslutande kommentarer

Vi har sett några användbara kommandon som bearbetar textfiler och hur man kan sätta samman enkla kommandon med rör för att lösa med komplicerade uppgifter. Detta fungerar tack vare att kommandona är gjorda så att de använder "standard-in" och "standard-ut" för in- och utmatning.

Nuförtiden är det förstas inte bara text man bearbetar i sin dator, utan även bilder och ljud, t ex. Tekniken med rör kan fortfarande användas. För bildbehandling finns t ex PNM-paketet. Ett kommando som konverterar en GIF-bild till en JPEG-bild med halva storleken skulle t ex kunna se ut så här:

```
giftoppm bild.gif | pnmscale 0.5 | cjpeg >bild.jpg
```

# Laborationer i kursmomentet Datoranvändning E1

<http://www.etek.chalmers.se/~hallgren/Eda/>

## Laboration nr 7: Programmering i Maple

1996, 1997 Magnus Bondesson  
1998 och 99-11-02 Thomas Hallgren

### 1 Introduktion

Syftet denna gång är att ge dig en viss inblick i vad programmering är och hur den går till samt vad vi kan uppnå med sådan. Vi arbetar i Maple men förhoppningen är att du skall få en allmän förståelse. Laborationen går utöver vad som nämns i Gula Boken. Jag försöker dock här att ta med alla erforderliga huvuddrag.

#### 1.1 Förberedelser

Innan du kommer till laborationen är det lämpligt att ha tittat på några sidor i Gula Boken, denna gång avsnitt 9.8 och 9.4.8. Läs också åtminstone avsnitt 2 i detta laborations-PM och fundera igenom några av de inledande Maple-uppgifterna hemma. Det är också lämpligt att **gå på föreläsningen**, speciellt för den som inte kan något om programmering sedan tidigare!

#### 1.2 Redovisning

Spara svaren på alla Maple-uppgifterna i ett och samma Maple-dokument. Spara ofta, så att inte allt går förlorat om Maple skulle krascha. Visa dokumentet för handledaren för att bli godkänd.

### 2 Kort om programmering

**Programmering** handlar om att i vid mening automatisera ett förfarande, dvs i stället för att manuellt genomföra ett antal åtgärder, så skriver vi ett **program** som genomför dem. I alla problemlösningsverktyg finns programmeringsmöjligheter, bl a av det skälet att användaren skall kunna lösa nya problem och inte bara de som verktyget har mer eller färdiga lösningar till.

Det finns en del begrepp som är centrala för all programmering:

- Variabel
- Styrkonstruktionerna räknarsnurra och villkorssnurra
- Styrkonstruktionen villkorligt utförande
- Funktioner/procedurer
- Parametrar

- Rekursion
- Konstruktioner som gör att man klarar sig utan t ex snurror ibland. I Maple finns t ex `seq`, `map`, `add`, `sum`, `mul`, `product` m fl.

Hur dessa ser ut i just Maple framgår av Gula Boken, men låt oss ändå samla huvud-  
dragen här. För detaljer hänvisas till boken.

**Räknar-snurrans form:**

```
for variabel from start by steg to slut do
    satser ;
od;
```

**Exempel**

```
# Beräknar summan 1+2+3+ ... + 10
s:=0;
for i from 1 to 10 do
    s:=s+i;
od;
```

Semikolonet efter *satser* behövs egentligen inte här eller i det följande och jag sätter inte alltid ut det i exemplen.

**Inmatning** av sådana här konstruktioner gör man i praktiken på något av de tre sätten (av de redigeringskäl som vi redovisat tidigare under kursen):

- Om den är mycket kort: På en rad.
- Om den är litet längre: Rad för rad med `SHIFT/RETURN`-tangenterna mellan raderna. Den sista raden avslutas med retur-tangenten varvid konstruktionen utförs.
- Om den är längre: Mata in den i något redigeringsprogram, t ex *NEdit*. Spara under lämpligt namn. Skriv i Maple: `read `filnamnet` ;` och konstruktionen utförs.

Från Maples synpunkt kan kod skrivas in hur som helst, t ex med allt på en rad eller ett ord per rad. Men för att koden skall bli läsvänlig mänskligt sett, bör man **strukturer** den så som vi visar i ovanstående och följande modeller.

**Villkors-snurrans form:**

**Exempel**

```
while villkor do
    satser ;
od;
```

```
# Beräknar summan 1+2+3+ ... + 10
s:=0;i:=1;
while i<=10 do
    s:=s+i;i:=i+1;
od;
```

Det finns också en kombinerad version som inleds med `for ... to slut while villkor do`.

**Villkorskonstruktionen** kan ta några olika former

```
if villkor then          if villkor then          if villkor1 then
    satser ;              satser ;                      satser ;
fi;                       else
                           satser ;                      elif villkor2 then
                           fi;                               satser ;
                           else
                           satser ;
                           fi;
```

## Exempel

```
# Bestämmer det största av två tal x och y, max är ett upptaget namn
if x>y then
    maxi:=x;
else
    maxi:=y;
fi;
```

En **funktionsdefinition** kan ges på ett par olika former. Den första har vi tidigare mött och kan användas när beräkningen inte behöver några hjälpvariabler.

```
funktionsnamn:=(parametrar)->uttryck i parametrarna;
```

Om det bara finns en parameter, behövs inga parenteser. Behövs hjälpvariabler (dessa införs i local-delen) får man ta till den allmännare formen

```
funktionsnamn:=proc(parametrar)
    local lokala_namn;
    satser;
end;
```

Definitionen "ekas", vilket kan hindras om man ersätter det avslutande semi-kolonet med ett kolon. I båda fallen sker anrop med

```
funktionsnamn(aktuella parametrar);
```

## Exempel

```
> f:=x->x^5;
      f := x -> x^5
> f(2);
      32
> f:=proc(x)
      x^5;
end;
      f := proc (x) x^5 end
> f(2);
      32
> f:=(x,n)->x^n;
      f :=(x, n) -> x^n
> f(2,4);
      16
> f:=x->if x<0 then -x else x fi;
f := proc (x) options operator, arrow; if x < 0 then -x else x fi end
> f(-3);f(2);
      3
      2
> f:=(x,y)->x^2+y^2;
```



```
f := (x, y) -> x^2+y^2
> f(1,2);
5
> summa:=proc(n)
    local s,i;
    s:=0;
    for i from 1 to n do
        s:=s+i;
    od;
end;
summa := proc (n) local s, i; s := 0; for i to n while true do s :=
s+i od end
> summa(10);
55
> summa:=n->if n=0 then 0 else n+summa(n-1) fi;
summa := proc (n) options operator, arrow; if n = 0 then 0 else
n+summa(n-1) fi end
> summa(10);
55
```

**Utskrifter** i styrkonstruktioner och funktioner kan vålla en del problem, se avsnitt 9.9 i Gula Boken. I korthet: Om du tycker att du får för många utskrifter, ändra systemvariabeln `printlevel` till 0 eller -1 före konstruktionen (återställ till standardvärdet 1 efteråt). Om du saknar en utskrift av något, framtvinga den genom att skriva `print(de saker du vill ha utskrivna)`. Känn dig inte tvingad att ta bort överflödiga utskrifter i uppgifterna.

### 3 Några erinringar angående Maple

Kommer du ihåg detta från Laboration nr 3?

- Varje kommando skall avslutas med `;` (eller `:` om man inte vill se Maples svar på kommandot).
- I Maple används `:=` för att göra definitioner. T ex betyder kommandot `x:=2` att variabeln `x` i fortsättningen står för talet 2.
- Vid redigering flyttar man sig upp i den tidigare textmassan genom att klicka med **vänstra musknappen** eller med **piltangenterna**. Vid utförandet av ett tidigare kommando (eventuellt redigerat) ersätts det tidigare svaret. Med musen eller piltangent kan man återvända till slutet.
- Undvik att trycka på returtangenten om något i textmassan är markerat; det markerade tas nämligen som ett kommando.
- Observera också att liksom i UNIX betyder i Maple stor och liten bokstav olika sak (oftast).

### 4 Uppgifter

Laborationstiden räcker säkert inte till för att göra allt. Uppgifterna bör göras i angiven ordning och \*-märkta uppgifter i första hand. Övriga uppgifter är frivilliga.

För att få roligare uppgifter kommer vi successivt att introducera ytterligare detaljfakta om Maple. Meningen är därmed inte att du skall känna dig tvingad att lära dig dessa utantill och de är inte huvudsyftet med uppgifterna, varför jag försöker ge så mycket hjälp som möjligt. Fråga även handledaren vid behov.

### **\*Uppgift 1. En funktion - repetition**

Du kommer väl ihåg hur man definierar funktioner i Maple? Skriv en Maple-funktion motsvarande  $f(x)=x^4-1$ . Kalla den t ex FUNK1. Bilda även derivatan:

```
FUNK1DER := D ( FUNK1 ) ;
```

### **\*Uppgift 2. En räknarsnurra**

Skriv satser som beräknar och skriver ut  $[x, f(x), f'(x)]$ , där  $f$  är funktionen i förra uppgiften, för  $x=0, 0.1, 0.2, 0.3, \dots, 0.9, 1.0$ .

### **\*Uppgift 3. Collatz problem med WHILE och IF**

Låt  $N$  vara ett positivt heltal större än 1. Om det är jämnt (kan avgöras med att man testat om  $N \bmod 2 = 0$ ), halvera det, i annat fall multiplicera det med 3 och lägg till ett. Om det nya talet är 1, är vi nöjda. Upprepa annars förfarandet med det nya. Det har praktiskt visat sig att man alltid når 1, men ingen har lyckats bevisa det. Exempel:  $N=5$  ger följden: 5,16,8,4,2,1. Skriv satser som givet  $N$  skriver ut följden i form av en sekvens eller lista.

### **Uppgift 4. Primtalstvillingar med WHILE och IF**

Två heltal  $N$  och  $N+2$  kallas primtalstvillingar om båda är primtal. Skriv satser som bestämmer det första tvillingparet som är större än 700. Gå igenom de tänkbara kandidaterna och bryt när det önskade paret hittats. Påminnes om att funktionen `isprime` avgör om ett tal är primtal eller ej.

### **Uppgift 5. Primtalstvillingar, forts. med FOR och IF**

Skriv satser som på formen `[17,19]` skriver ut samtliga primtalstvillingar mindre än 2000 och även räknar antalet sådana.

### **\*Uppgift 6. En funktion som ger sista siffran i ett positivt heltal**

Skriv en funktion `Sista(N)` som ger sista siffran i heltalet  $N$ . Sista siffran fås som heltalsresten vid division med 10, vilket Maple ger med  $N \bmod 10$ .  
Exempel: `Sista (184)` ger 4.

### **\*Uppgift 7. En funktion som ger första elementet i en lista**

Någon sådan behöver vi inte eftersom första elementet i en sekvens eller lista  $L$  kan fås med `L[1]`, men gör ändå en sådan funktion `First(L)`. Ex: `First([8,1])` ger 8.

### Uppgift 8. En funktion som ger de två första elementen i en lista

Skriv en funktion `FirstPair(L)`, vars värde är listan bestående av de två första elementen i listan `L`.

### \*Uppgift 9. En funktion som kastar om elementen i en tvåelementlista

Skriv en funktion `Change(L)`, vars värde är en lista med elementen `L[1]` och `L[2]` omkastade. Text skall `Change([7,9]);` ge `[9,7]`.

### \*Uppgift 10. En funktion som summerar elementen i en lista av tal

Skriv en funktion `Summa(L)`, vars värde är summan av alla elementen (som förutsättes vara tal) i `L`. Text `Summa([3,1,9]);` skall ge 13. **Tips:** använd den inbyggda funktionen `add`, eller gör en for-snurra.

### \*Uppgift 11. En egen funktion

Gör en egen funktion som beräknar N-fakultet, dvs fungerar som den inbyggda `factorial`. Funktionen skall alltså beräkna  $1*2*3*...*N$ . **Tips:** använd den inbyggda funktionen `mul`, eller gör en for-snurra.

### \*Uppgift 12. En funktion som letar i en lista

Skriv en funktion `Leta(L,value)`, som letar igenom listan `L` efter värdet `value` och returnerar motsvarande platsnummer (0 om värdet inte finns med). Text `Leta([7,5,13,8],13);` ger 3. **Tips:** använd en while-snurra. Funktionen `nops` kan beräkna antalet element i en lista.

### Uppgift 13. En funktion behöver inte beräkna ett egentligt värde

Skriv en funktion `Rita(f)`, som ritar upp funktionen  $f(x)$  på intervallet  $[-1,1]$ . Text skall `Rita(x->x^2)` rita en del av en parabel.

### \*Uppgift 14. Automatisera Newtons metod

Newtons metod för att bestämma ett nollställe till en funktion  $f(x)$  består väsentligen i Maple i man upprepar

$$x := x - f(x) / D(f)(x);$$

ett antal gånger. Först måste vi naturligtvis ha definierat funktionen och gett `x` ett startvärde. Automatisera beräkningarna med en snurra. Skriv för varje iterationsvärde ut dels det nya `x`-värdet, dels skillnaden mot det gamla. **Tips:** `printlevel:=0` före snurran och `print([x,x-old])` ger precis de önskade utskriftena.

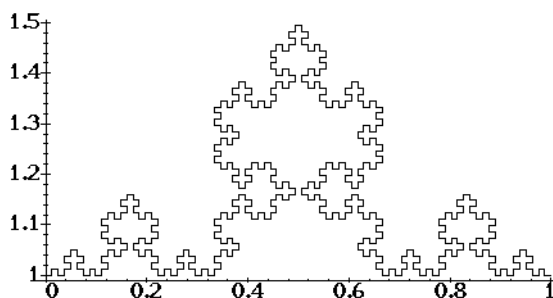
### Uppgift 15. En funktion som bestämmer ett nollställe till en godtycklig funktion

Stoppa nu in det du gjorde i förra uppgiften i en Maple-funktion `Newton(f,x0)` som beräknar ett nollställe till  $f(x)$  i närheten av  $x0$  (eventuellt komplext; metoden fungerar utan ändringar precis lika bra då; för att komplexa nollställena måste man starta med komplext  $x0$ ). Prova den på funktionen  $f(x)=x^4-1$  som ju har de fyra nollställena  $1, -1, i, -i$ .

Låt beräkningen avbrytas då skillnaden mellan de två senaste värdena är liten (t ex  $\text{abs}(x-x_{\text{old}}) < 0.00001$ ), men gör maximalt 20 iterationer (flera villkor kan kombineras med and).

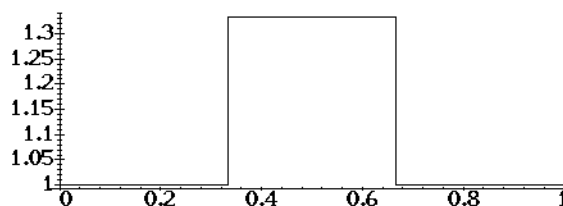
### Uppgift 16. Välkommen till fraktalernas värld

Kurvan här intill är ett exempel på en sk fraktalkurva, ett begrepp som myntades av den fransk-amerikanske matematikern Mandelbrot i 1975. Fraktal betyder bruten och det är ju synbart en egenskap som kurvan har. Kurvan börjar i punkten (0,1) och slutar i (1,1). Det är inte ovanligt att man i programmeringskurser med grafikinslag tar upp ritandet av sådana. Dels därför att det kan vara skojigt, men framför allt för att rekursion är en



behändig teknik i sammanhanget. Det går bra även i Maple, men man får tänka litet annorlunda än i ett traditionellt programspråk och tidsåtgången blir betydligt större.

Låt oss först förklara idén. Den ovan ritade kurvan är 4:e generationen i en familj vars 0:te generation utgörs av ett rakt horisontellt streck från (0,1) till (1,1) och vars 1:sta generation är kurvan (kallad generatoren eftersom den bestämmer de följande generationernas



utseende). Nästa generation, den andra, får vi genom att ersätta de fem streck som bygger upp denna kurva med en förminskad version av generatoren, precis som vi fick den första generationen genom att ersätta det ursprungliga strecket med generatoren. Låt  $P_0$  och  $P_1$  vara start- och slutpunkt på ett generatorsegment och låt  $P_2, P_3, P_4$  och  $P_5$  vara de mellanliggande brytpunkterna (markerade för hand i figuren ovan). Om vi nu vill rita ett visst generationsnummer  $N$ , ritar vi helt enkelt närmast lägre nummer mellan  $P_0$  och  $P_2$ , mellan  $P_2$  och  $P_3$ , mellan  $P_3$  och  $P_4$ , mellan  $P_4$  och  $P_5$  och till sist mellan  $P_5$  och  $P_1$ . Detta är en rekursiv process, som lätt låter sig formuleras. En mindre svårighet är att beräkna koordinaterna för punkterna, så därför ger jag dem här (vi representerar punkter med en tvåelementslista  $P=[x,y]$ , dvs  $P[1]$  är x-koordinaten):

```
dx:=(P1[1] - P0[1])/3: dy:=(P1[2] - P0[2])/3:
P2:=[P0[1]+dx, P0[2]+dy]: P3:=[P2[1]+dy, P2[2]+dx]:
P5:=[P1[1]-dx, P1[2]-dy]: P4:=[P5[1]+dy, P5[2]+dx]:
```

Skriv nu en rekursiv funktion  $fract(N,P_0,P_1)$  som genererar en sekvens  $S$  av samtliga brytningspunkter för kurvan i den  $N$ :te generationen,  $N \geq 0$ . Kurvan för  $N=4$  kan sedan ritas med

```
plot([ fract( 4,[0,1],[1,1] )], scaling=constrained);
```

Jag påminner om att två sekvenser  $S1$  och  $S2$  kan sättas ihop till en enda med  $S1,S2$ .

### Uppgift 17. Beräkna $\pi$ med många siffror

En nyligen (under 1980-talet) upptäckt snabb metod för beräkning av talet  $\pi$  med massor av decimaler lyder som följer

1. Sätt  $y_0 = \sqrt{2} - 1$  och  $\alpha_0 = 6 - 4\sqrt{2}$

2. Upprepa för  $n=0,1,2,3,\dots$

2.1.  $y_{n+1} = \frac{1-b}{1+b}$ , där  $b = (1-y_n^4)^{1/4}$

2.2.  $\alpha_{n+1} = (1+y_{n+1})^4 \alpha_n - 2^{2n+3} y_{n+1} (1+y_{n+1}+y_{n+1}^2)$

Då närmar sig  $p=1/\alpha_n$  talet  $\pi$  mycket snabbt när  $n$  växer; antalet riktiga siffror mer än fyrdubblas per upprepningssteg. Verifiera detta praktiskt med Maples hjälp enligt:

1. Sätt antalet siffror som Maple arbetar med till t ex 500 med `Digits:=500;`.
2. Vi behöver tre variabler  $y$ ,  $a$  och  $p$  för  $y_n$ ,  $\alpha_n$  och  $p$  ovan. Ge  $y$  och  $a$  startvärden ( $n=0$ ).
3. Beräkna i en räknarsnurra med 5 varv  $y = y_n$  och  $a = \alpha_n$  samt  $p$ , för  $n=1,2,3,4,5$ . Beräkna med `evalf(p-Pi)` felet och skriv i varje varv ut  $p$  och felet. Notera hur felets storleksordning avtar.
4. Återställ antalet arbets-siffror med `Digits:=10;`

## 5 Avslutande kommentarer

Detta är inte en programmeringskurs, så meningen med denna laboration är som sagt bara att ge en liten inblick i vad programmering kan innebära.

### 5.1 Information på webben

Hemsidan för kursen Programmeringsteknik E2 finns på adressen

<http://www.cs.chalmers.se/Cs/Grundutb/Kurser/e2pt/>

Andra kurser som ges av Institutionen för Datavetenskap finns på

<http://www.cs.chalmers.se/Cs/Grundutb/Kurser/>

För den som är intresserad av hur program ser ut i olika programspråk rekommenderas sidan

<http://www.ionet.net/~timtroyr/funhouse/beer.html>

där ett och samma program finns översatt till över 200 olika programspråk.

# Laborationer i kursmomentet Datoranvändning E1

<http://www.etek.chalmers.se/~hallgren/Eda/>

## Laboration nr 8: Matteverktyget MATLAB

1992-1997 Magnus Bondesson

Uppdatering 1998 och 99-11-09 Thomas Hallgren

**Sista godkännandedag: torsdag 99-12-09 kl 17.00**

### 1 Introduktion

Syftet med denna laboration är att ge dig en första praktisk kontakt med matematikverktyget MATLAB. Precis som i Laboration nr 3 (som handlade om Maple) avser vissa av uppgifterna dessutom att antyda hur dessa verktyg kan användas för att ge ökad förståelse kring något fenomen.

#### 1.1 Förberedelser

Som förberedelse är det lämpligt att ha läst igenom MATLAB-kapitlet i Gula Boken (hoppa som vanligt över det som verkar konstigt). En introduktion till MATLAB ges också på föreläsningen.

Det är lätt att blanda ihop Maples beteckningar med MATLABs. Observera speciellt att definitioner i MATLAB görs med "=" och att kommandon normalt *inte* avslutas med ";" (";" använder man om man inte vill se MATLABs svar på ett kommando!).  $\pi$  heter `pi` och inte som i Maple `Pi`, etc.

MATLAB arbetar till skillnad mot Maple inte i ett eget fönster, utan i ett vanligt terminalfönster. Piltangenterna fungerar ungefär som i *tsch* (man kan t ex få tillbaka tidigare inmatade kommandon med uppåtpilen). Man kan markera gammal text med musen och klistra in den i nya kommandon. Figurer visas i egna fönster (man kan ha flera, men normalt återanvändes ett enda automatiskt).

#### 1.2 Redovisning

När man jobbar i Maple är det lätt att spara allt man gör. Tyvärr är det inte lika enkelt i MATLAB, eftersom det körs i ett vanligt terminalfönster. För att bli godkänd måste man i alla fall

- skriva ner alla kommandon som behövs för att lösa uppgifterna. Detta kan man göra på olika sätt. Ett sätt som verkar ganska enkelt är markera text man vill spara i MATLABs terminalfönster och klistra in den i ett *NEdit*-fönster. Man markerar text genom att dra med musen. En hel rad kan markeras genom att man trippelklickar på den. Man kan klistra in text i *NEdit* genom att klicka med mittenknappen.

- spara figurer som ritas i uppgifterna när så anges. Detta kan man göra från figurfönstrets knapprad eller meny.

och visa allt för handledaren.

## 2 Uppgifter

### Uppgift 1. Läs i `cth.etek.e1`

Kanske har Thomas kommit på något lättare sätt att spara allt man gör i MATLAB, så man slipper klippa och klistra så mycket. Titta i nyhetsgruppen `cth.etek.e1` efter ett meddelande om detta.

### Uppgift 2. Starta MATLAB

Det är lämpligt — men inte nödvändigt — att först skapa en katalog (mapp) `matlab`, vilket du gör med UNIX-kommandot `mkdir matlab`. Starta sedan MATLAB med `tex` ett programmeny-kommando.

### Uppgift 3. En inblick i MATLABs värld

Innan vi börjar med uppgifter som illustrerar de mer grundläggande funktionerna i MATLAB kan det kanske vara kul att ta en titt på lite mer avancerade exempel för att få en känsla för vad man kan använda MATLAB till.

Till MATLAB hör en massa demonstrationsprogram, som är skrivna i MATLAB (källkoden är tillgänglig). Du kommer in i demonstrationsdelen genom att ge kommandot `demo`. Gör det och pröva `Visualization/2D Plots`, `Visualization/3D Plots`, `Gallery/Hoops`, `Games/Minesweeper` och `Visualization/Lorenz attractor animation`.

### Uppgift 4. Vektorer, ritning och redigering

Vi skall rita upp kurvan  $\sin(x)$  på intervallet  $[0,16]$ .

- Bilda en vektor  $x = [0, 1, 2, \dots, 16]$  med litet skrivarbete.
- Bilda en vektor  $y$  med motsvarande sinus-värden.
- Rita kurvan med `plot(x,y)`.
- Resultatet blev inte alldeles lyckat eftersom vår diskretisering var väl grov. Ändra med `piltangenternas` hjälp (se sid 169) så att  $x = [0, 0.1, 0.2, \dots, 16]$  i stället samt — även detta med `piltangenterna` — bilda motsvarande vektor  $y$  och rita på nytt.
- Spara figuren!

### Uppgift 5. Addition och slumpstal

Rita upp kurvan  $\sin(x)$  på samma intervall som i Uppgift 4, men nu störd med slumpmässigt brus.

- Låt bruset vara slumpmässiga tal intervallet (0.0,0.4) som adderas till sinusvärdena. Vektorn  $x$  har du kvar sedan förra uppgiften, så det räcker att bara skriva ett nytt *plot*-kommando. **Ledning:** Slumptalsfunktionen *rand* beskrivs i Gula Boken, sid 170. Förmodligen är  $rand(size(x))$  ganska användbart.
- Låt bruset vara slumpmässiga tal i intervallet (-0.2,0.2) istället. Spara figuren!

### Uppgift 6. Funktioner som arbetar på vektorer

Betrakta uttrycket  $\frac{4}{1+x^2}$ .

- Beräkna värdet för  $x=0.5$ .
- Beräkna i ett svep uttryckets värde för  $x = 0, 0.1, 0.2, \dots, 1.0$ .
- Rita upp motsvarande kurva från  $x = 0$  till  $x = 1$ .
- Spara figuren!

### Uppgift 7. Rötter

Bestäm rötterna till 16:e-gradsekvationen  $17x^{16} + 16x^{15} + \dots + 3x^2 + 2x + 1 = 0$ . Rita upp dem i ett diagram (markera varje rot med + eller \*). **Ledning:** Funktionen *roots* beskrivs i Gula Boken, sid 172. Om du händelsevis hann med Uppgift 23 i Laboration nr 3, notera hur mycket enklare detta problem löses i MATLAB än i Maple.

Spara figuren!

### Uppgift 8. Komplexa tal

- Låt  $z = 1 + 2i$ . Beräkna uttrycket  $z^3 + z^5$ .
- Beräkna sedan i ett svep samma uttryck för  $z = 1 + 2i, 1 + 3i, \dots, 1 + 9i$ .

### Uppgift 9. Egna funktioner. Integration.

- Tag med hjälp av *help* eller *helpdesk* reda på vad verktygslådefunktionen *quad* gör (läs bara de 5 första raderna). Beräkna sedan en approximation till integralen  $\int_0^{\pi/2} \cos x dx$  och jämför med det exakta värdet 1. **Notera** att funktionens namn måste stå inom apostrofer i anropet av *quad*.
- Skriv en MATLAB-funktion *first*, dvs gör en m-fil (sid 183) `first.m` med ditt favoritredigeringsprogram (t ex *NEdit*), som beräknar  $\frac{4}{1+x^2}$  för en **godtycklig vektor**  $x$ . Spara lämpligen i katalogen `matlab` som du skapade inledningsvis. En sådan funktion har formen



```
function y=first(x)
y=?????
```

Testa den och verifiera att den ger rätt resultat.

- c. Beräkna nu med *quad* en approximation till integralen  $\int_0^1 \frac{4}{1+x^2} dx$  och jämför med det exakta värdet som ju är  $\pi$ .

### Uppgift 10. Ett besök till i fraktalernas underbara värld

Låt oss välja en punkt  $(x_0, y_0)$  i  $x, y$ -planet och sedan bilda nya punkter enligt formeln,

$$\begin{cases} x_{n+1} = 1 - y_n + |x_n| \\ y_{n+1} = x_n \end{cases}$$

för  $n = 0, 1, 2, \dots$ . Var hamnar dessa punkter? Följande funktion i form av en m-fil gör 1000 iterationer utifrån en startpunkt och ritar samtliga punkter. Skriv in funktionen (texten som inleds med % kan du strunta i) med valfritt redigeringsprogram och spara den som filen *ginger.m* i din katalog *matlab*.

```
% GINGER(x,y) draws PEPPARKAKSGUBBEN if e g x=-1.3,y=2
function ginger(x,y)
clf                                %erase previous figure and forget hold
axis([-10,10,-10,10])             %our coordinate system
plot([-10,10],[0,0])               %draw x-axis
hold on                             %never erase,but autoscale still in effect
plot([0,0],[-10,10])               %draw y-axis
for i=1:1000                         %iterate and plot 1000 times
    x1=1-y+abs(x);y=x;x=x1;
    plot(x,y, '.')
end
```

Anropa den med *ginger(-1.3,2)*. Experimentera med andra startvärden. Resten är klar överkurs. Ändra i funktionen *abs(...)* till *sqrt(abs(...))* eller *sin(...)*. Gör eventuellt om funktionen så att startpunkt väljs interaktivt med musen (jämför med Uppgift 18).

Spara figuren!

### Uppgift 11. WWW-hjälp

Dokumentationen till MATLAB finns även som WWW-sidor (se sid 177). Tag den vägen reda på information om *ezplot*-kommandot.

## 3 Frivilliga extrauppgifter

Några av de följande uppgifterna kräver kanske lite grundläggande kunskaper i programmering.

### Uppgift 12. Vektorer och Lissajous-kurvor

Rita upp kurvan  $x = \cos 4t$ ,  $y = \sin(3t)$  på  $t$ -intervallet  $[0, 2\pi]$ . Konstanten  $\pi$  finns fördefinierad.

### Uppgift 13. Rotvandring

Studera hur rötterna vandrar när nolltegrads-termen i förra uppgiften avtar från 1 till  $-1$ , genom att i samma diagram rita in rötterna (med markering `.`) när nolltegradstermen är 1, 0,9, 0,8, ...,  $-0,9$ ,  $-1,0$ . Gör först *hold on* för att få ritning i samma diagram och ändra sedan nolltegradstermen successivt i en FOR-snurra (eller arbeta manuellt). Vad händer med de två rötter som ursprungligen låg längst till vänster i diagrammet? Hur många reella rötter har utgångsekvationen respektive sluttekvationen? Gör *clf* (av Clear Figure) eller *hold off* innan du fortsätter med en ny uppgift.

### Uppgift 14. Bild från MATLAB fogas in i FrameMaker eller webbsidor

Spara diagrammet från någon av de två föregående uppgifterna som en PostScript-fil MATTE.epsi (se sid 180). Kontrollera resultatet med t ex

```
ghostview MATTE.epsi.
```

**Anm.** Från Laboration nr 4 vet vi hur bilder kan importeras till FrameMaker. Även PostScript-bilder kan importeras.

Vill man skapa bilder till webbsidor är det lämpligt att spara i något av formaten PNG, JPEG eller GIF. Kommandot *print -djpeg filnamn* sparad bilden i JPEG-format och kommandot *print -dpng filnamn* sparar i PNG-format. Man kan inte skapa GIF-bilder direkt, men i Gula Boken (sid 181 överst) står det hur man kan göra.

Kommandot *help print* ger aktuell och detaljerad information.

### Uppgift 15. Komplexa vektorer

Låt  $z$  vara en vektor av punkter som ligger på enhetscirkeln i det komplexa talplanet. En sådan vektor kan bildas t ex så här:

```
t=0:0.1:2*pi; z=cos(t)+i*sin(t).
```

Var hamnar då de punkter som utgörs av  $w = \frac{1-z}{1+z}$ ? Vill du visualisera resultatet är det lämpligt att förfara enligt (>> är redotecknet som ej skall skrivas):

```
>> axis([-10,10,-10,10]); % Skala på axlarna
>> plot(z, 'o')           % Rita punkterna på cirkeln
>> hold on                % Rita utan att sudda föregående
>> plot(uttryck för w, 'o') % Rita de nya punkterna
>> clf                    % Innan du går vidare
```

Svar: \_\_\_\_\_

### Uppgift 16. Summation

Beräkna följande summor numeriskt. (**Ledning:** *sum*, se sid 171.)

- a.  $\sum_{k=1}^{60} \frac{1}{k^4}$ . Jämför gärna med summan  $\frac{\pi^4}{90}$  för motsvarande oändliga serie.
- b.  $\sum_{k=1}^{30} \sin \frac{1}{k}$ . Svar: \_\_\_\_\_

### Uppgift 17. Egna funktioner. Ekvationslösning.

- a. Tag med hjälp av *help* reda på vad verktygslådefunktionen *fsolve* gör. Beräkna sedan en approximation till det nollställe till  $\cos x$  som ligger i närheten av 1.5. Jämför med det exakta värdet som ju är  $\pi/2$ .
- b. Funktionen  $5xe^{-x} - 1$  har två reella nollställen på intervallet  $[0,5]$ . Visa det genom att rita upp funktionen.
- c. Skriv en MATLAB-funktion *second*, dvs gör en m-fil *second.m* utanför MATLAB med ett redigeringsprogram, som beräknar funktionsvärdena för en godtycklig vektor. En sådan funktion har formen

```
function y=second(x)
y=?????
```

- d. Beräkna nu med *fsolve* det största av de två nollställena i b. Vad blev det?
- \_\_\_\_\_

### Uppgift 18. Egna funktioner. Interaktion.

Skriv en MATLAB-funktion *inter*, dvs gör en m-fil *inter.m*, som låter dig markera 10 punkter i ritfönstret (som förutsätts finnas framme) och när en punkt markerats förbinder den med närmast föregående (utom för den första). En sådan funktion har formen

```
function inter
[xold,yold]=ginput(1);
for i=1:9
    ???
end
```

där ??? står för maximalt 4 rader kod.

Gör *clf* (suddar ritfönstret) och *hold on* och testa sedan din funktion.

### Uppgift 19. 3D-ritning

Hitta på en trevlig funktion  $f(x,y)$  av två variabler. Skriv den som en m-fil och rita upp den med specialfunktionen *myplot3d* på ett uttryck (se sid 175 och sid 186) eller gör stegen i den funktionen.

### Uppgift 20. Även solen har sina fläckar

Följande lilla problem i MATLAB man kan lätt hamna i och bli förundrad över. Säg att vi vill rita upp roten till ekvationen  $x^2 - 1 = 0$ . Vi vet vid det här laget att det kan åstadkommas med *plot(roots([1,0,-1]), '\*')*. Pröva det. Konstatera att resultatet är käpprätt åt skogen. Förklaringen kan du läsa dig till med *help plot* (andra stycket). Det egentliga skälet är att man givit *plot* olika betydelse beroende på vad första parametern är. Och det är uppenbarligen inte bra. Det här problemet dyker upp när samtliga rötter är reella. En nödlösning är att lägga till ett litet rent imaginärt tal (och hoppas att inte alla rötterna har en imaginärdel som är lika med -detta tal), dvs *plot(roots([1,0,-1])+0.001\*i, '\*')*. Förvissa dig om att detta fungerar.

### Uppgift 21. Zooma

MATLAB har vissa interaktiva inslag som saknas i Maple (och vice versa för den delen). Rita upp kurvan från Uppgift 17b. Tag med *help zoom* reda på vad *zoom*-kommandot utträttar. Tag sedan med dess hjälp reda på ett av nollställena med litet bättre noggrannhet än vad den ursprungliga figuren tillåter.

### Uppgift 22. Rotvandring med hemmagjord funktion

De flesta har nog inte hunnit med någon av de tidigare förekommande rotvandring-problemen. Själva grundidén med denna typ av uppgift är att man vill visualisera hur ändringar i indata påverkar resultaten i fall där beroendet är komplicerat. T ex vilka rötter som påverkas mycket.

Mata in m-funktionen nedan (kommentarerna kan du ju strunta i) och kalla filen *animate\_roots.m* (eller kopiera den från *~hallgren/Intro/*). Sätt dig även in i hur den fungerar.

```
function animate_roots(p,i,a)
    % p is a vector with the coefficients of a polynomial as for roots
    % i is an order, e g i=0 means the zero order coefficient
    % a is a vector with values for that coefficient
    % animate_roots will draw the roots for the different values of a
    % in a common window
    % the user has to press any key between the cases
    % E g animate_roots([4,3,2,1],1,2:0.1:3);
    % will draw the roots when the first order coeff varies from 2 to 3
    clf
    hold off
    for var=a
        p(length(p)-i)=var;
        r=roots(p);
```

```
        plot(r, '+');  
        hold on;  
        pause  
    end  
    hold off
```

Pröva den sedan för det fall som nämns bland kommentarerna och för fallet i Uppgift 13.

### Uppgift 23. Program med knappar och menyer och sånt

Tydligt kan man inom MATLABs ram skapa trevliga interaktiva program. Vi skall belysa den tekniken litet grand. Mata in följande som en m-fil `USER.m` (eller kopiera från den vanliga platsen). Förklaringar har jag förhoppningsvis presenterat under föreläsningen.

```
% This is a short m-file with commands only, they are executed  
% when the name of the file (.m excluded) is written as a command  
  
% Create a window for graphics  
fig = figure;  
  
% Create a button control  
h1 = uicontrol('Position',[10,20,200,60]);  
set(h1,'Callback','s=33; s+2');  
set(h1,'String','Tryck här');  
  
% Create another button control for ending the experiment  
h2 = uicontrol('Position',[210,20,200,60]);  
set(h2,'Callback','delete(fig)');  
set(h2,'String','Avsluta');  
  
% Create a scroll-bar control; the value is written  
h3 = uicontrol('Position',[150,90,200,60]);  
set(h3,'Callback','s=get(h3, 'value')');  
set(h3,'Style','slider');
```

Anropa den med `USER`. Pröva de tre styrorganen. Notera att hissen ger ett värde  $s$  mellan 0 och 1. Modifiera sedan den så att kurvan  $y = e^{sx} - s$  eller eventuellt kurvorna  $y = e^{sx}$  och  $y = x$  ritas upp på intervallet när man dragit i hissen. Använd alternativt verktyget *guide* för att ändra användargränssnittet.

## 4 MATLAB på webben

Företaget som gör MATLAB har webbadressen [www.mathworks.com](http://www.mathworks.com). Där kan man hitta en del reklam för MATLAB. Man kan hitta mer nyttig information genom att söka efter MATLAB i YAHOOs webbkatalog på [www.yahoo.com](http://www.yahoo.com).

## Laborationer i kursmomentet Datoranvändning E1

<http://www.etek.chalmers.se/~hallgren/Eda/>

### Laboration nr 9: Textredigering, Unix och praktik

1998 och 99-11-23 Thomas Hallgren

**Sista godkännandedag: torsdag 99-12-09 kl 17.00**

## 1 Introduktion

Syftet med denna laboration är att ge ytterligare praktiska färdigheter i att arbeta med Unix-datorer. Det ingår övningar på att organisera filer och att redigera textfiler. En övning med ett annat tema handlar om [workingday.com](http://workingday.com), en webbtjänst som kan göra det lättare att hitta praktikjobb.

Kommandon som dyker upp är: `gzip`, `gunzip`, `cjpeg`, `xv`, `tar`, `mkdir`, `mv`, `rm`, `du`.

Begrepp som dyker upp är: filformat, kompression, omdirigering, paket (arkivfiler), rör, filnamnsmonster.

### 1.1 Förberedelser

Läs i Gula Boken om filnamnsmonster (avsnitt 1.16.2). Repetera omdirigering och rör (avsnitt 1.16.1, Laboration nr 6). Bläddra igenom uppgifterna och läs speciellt Uppgift 6 i förväg!

### 1.2 Redovisning

Skriv svar på frågorna i häftet. Spara filer som skapas under arbetet. Visa allt för handledaren för att bli godkänd. Uppgift 6 godkänns på annat sätt.

## 2 Uppgifter

Som tidigare markerar symbolen  i uppgifter med flera delar inledningen på ett stycke med en konkret arbetsuppgift. Dessförinnan kan finnas motiverande eller förklarande text.

### Uppgift 1. Elektronisk deltagarlista

Som förhoppningsvis alla redan vet, så kan man se på webben vilka intro-grupper som är godkända på vilka laborationer. För att kunna rapportera vilka som är godkända på kursen måste ett rättningsprotokoll med namn och personnummer skapas. Denna information finns i och för sig på anmälningslistan som alla skrev på i början

av kursen, men för att bespara kursansvarig och annan personal en massa tråkigt arbete, och minska risken för fel, skulle det vara bra om alla fyllde i samma information på den *elektroniska deltagarlistan* också.

För att skriva upp sig på elektroniska deltagarlistan använder man kommandot

```
elperson
```



Det är bara två uppgifter man behöver fylla i: sitt personliga konto, och sitt personnummer. Gör det nu!

När du har fyllt i dessa uppgifter kan du även använda kommandot

```
elperson godkänd
```

när du är *inloggad på ditt personliga konto*, för att kontrollera vilka laborationer du är godkänd på, om du har uppfyllt alla krav för att bli godkänd på kursen, och om du är rapporterad som godkänd på kursen.

## Uppgift 2. Komprimering av filer

Bild och ljud är exempel på information som kan ta mycket plats att lagra om man inte använder någon form av datakompression. Det finns både generella kompressionsmetoder, som fungerar på alla typer av filer, och kompressionsmetoder skräddarsydda för en viss typ av filer. De generella kompressionsmetoderna är exakta, dvs all information bevaras, även om den lagras på ett kompaktare sätt. De skräddarsydda kompressionsmetoderna kan ta bort information som är oviktig, t ex detaljer ur bilder/ljud som är svåra att uppfatta med blotta ögat/örat.

Ett vanligt generellt komprimeringsprogram i Unix är `gzip`. Filer komprimerade med `gzip` brukar ha namn som slutar på `.gz`. Det vanliga sättet att använda `gzip` är att skriva `gzip filnamn`, men `gzip` fungerar även ihop med omdirigering och rör. Filer komprimerade med `gzip` kan dekomprimeras med kommandot `gunzip`. Tag man `gzip` för detaljerad beskrivning!



Prova `gzip` på några filer och fyll i tabellen (om du inte redan har filerna kan du kopiera dem från `~hallgren/Intro/`):

Fil	Ursprunglig storlek (s0)	Komprimerad storlek (s1)	Kompressionsfaktor (s0/s1)
Berling.txt			
Propaganda.txt			
blomvas.ppm			

PPM är ett enkelt filformat, utan kompression, för bilder. JPEG är ett filformat för bilder med inbyggd, skräddarsydd kompression. Filerna brukar ha namn som slutar på `.jpg`.

Kommandot `cjpeg` konverterar från PPM-formatet till JPEG. Resultatet matas ut på standard-ut, så man får använda omdirigering för att lagra det i en fil. En JPEG-fil kan

konverteras tillbaka till PPM-formatet med kommandot `djpeg`. Tag man `cjpeg` respektive man `djpeg` för detaljerad beskrivning!

- Från den föregående uppgiften bör du nu ha kvar `blomvas.ppm.gz`. Dekomprimera den med `gunzip` och konvertera den till JPEG-format med `cjpeg`. Hur stor blev JPEG-filen? \_\_\_\_\_ Vad blir kompressionsfaktorn? \_\_\_\_\_

Titta på bilderna `blomvas.ppm` och `blomvas.jpg` med kommandot `xv`. Har kompressionen gett någon märkbar skillnad i bildkvalité?

### Uppgift 3. Många filer i en fil

Ofta när man hämtar hem program eller andra saker från Internet får man en fil, som i själva verket är ett *paket*, som innehåller många filer (själva programmet, dokumentation, extra tillbehör, mm). Paket, som också kan kallas *arkivfiler*, måste "packas upp" på något sätt innan man kan använda dem. Även i andra sammanhang kan det vara praktiskt att använda paket av filer, t ex om man ska skicka en samling filer till en kompis via elektronisk post.

I Windows på PC används ofta programmet *WinZip* för att hantera paket av filer. På Macintosh är *StuffIt* vanligt.

I UNIX används ofta kommandot `tar` för att skapa paket. (`tar` är en förkortning för *tape archive*, eftersom programmet från början var avsett att användas för arkivering av filer på magnetband.) Det används på följande sätt:

<code>tar cf paket.tar fil1 fil2 ...</code> <code>tar cf paker.tar katalog</code>	(c=create) Med <code>cf</code> skapas ett nytt paket <code>paket.tar</code> är namnet på paketet och resten är namn på filer som ska stoppas i paketet. Om något av filnamnen i själva verket är en katalog stoppas katalogen och alla filer i den in i paketet.
<code>tar af paket.tar fil1 fil2 ...</code>	(a=add) Med <code>af</code> kan man lägga till fler filer eller kataloger sist i ett paket
<code>tar tf paket.tar</code> <code>tar tvf paket.tar</code>	(t=table of contents) Med <code>tf</code> får man en lista över vilka filer som ingår i ett paket. Med <code>tvf</code> får man en detaljerad lista, av samma slag som man får med <code>ls -l</code>
<code>tar xf paket.tar</code>	(x=extract) Med <code>xf</code> kan man packa upp ett paket.

Tag man `tar` för detaljerad beskrivning!

**Anm.** Olika "packarprogram" använder olika filformat för paket. Kommandot `tar` känner bara till sitt eget format. Kommandot `unzip` kan användas för att packa upp filer i ZIP-formatet, som t ex *WinZip* använder.

- Filen `~hallgren/Intro/oordning.tar` är ett tar-paket. Tag en titt på innehållet med `tar tf`, eller kanske `tar tvf`. Hur många filer innehåller paketet (tips: om du inte orkar räkna själv, använd `tar tf`, ett rör och `wc`)? \_\_\_\_\_



- Hur många kataloger skapas om man packar upp paketet? \_\_\_\_\_

#### Uppgift 4. Att hålla ordning bland sina filer

När man arbetar med en dator blir det väl lätt så att man efter hand samlar på sig fler och fler filer. Det är då förstås lämpligt att skapa kataloger för filer som hör ihop. Filen `oordning.tar` i förra uppgiften är ett exempel hur det kan bli om man inte gör detta.

- Packa upp `oordning.tar` och läs filen `HELP`, där det står hur filerna ska ordnas. Använd kommandona `mkdir`, `mv` och `rm` för att ordna filerna på det sätt som anges.

#### Uppgift 5. Många filer i en fil sparar plats

- Packa de ordnade filerna från förra uppgiften till en fil som heter `ordnat.tar`. Komprimera den med `gzip`. Hur stor blev filen? \_\_\_\_\_
- Med kommandot `du katalog` kan man mäta hur stor plats filerna i en katalog tar. Hur stor plats tar filerna i katalogen som du nyss packade ihop? \_\_\_\_\_
- Nu när vi har ett paket med alla filer kan vi ta bort katalogen de låg i. Gör det! Det går att ta bort en katalog och hela innehållet med ett kommando. Hur gör man?

#### Uppgift 6. Textredigeringstävling

Tips: läs igenom hela uppgiften innan du gör något!

I kursen E2 Programmeringsteknik kommer ni under laborationerna att skriva program i något programspråk. Detta gör man med ett textredigeringsprogram, t ex *NEdit* eller *Emacs*. Det vanliga när man skriver program är att de inte fungerar första gången man provkör dem, utan det krävs ofta en serie med omväxlande justeringar och provkörningar innan programmet fungerar som det ska. För att detta arbete inte ska bli alltför tidsödande krävs viss vana att använda textredigeringsprogram. Denna uppgift är tänkt att ge lite sådan vana.

Uppgiften är att med hjälp av valfritt textredigeringsprogram rätta ett antal fel i en textfil på så kort tid som möjligt. Tillvägagångssättet är som följer:

- För att få en fil att redigera kör man kommandot

```
textred ny >filnamn
```

där *filnamn* kan bytas mot valfritt filnamn. Datorn genererar då en ny textfil med ett antal fel och noterar tidpunkten då filen lämnades ut. Detaljerad beskrivning av vilka ändringar som ska göras finns först i filen.

- Man rättar felen och kör sedan kommandot

```
textred svar filnamn
```

där *filnamn* är samma filnamn som tidigare. Datorn kontrollerar då, med pedantisk noggrannhet, att alla fel är rättade och att inga andra ändringar gjorts i filen. Om filen inte är korrekt kan man gå igenom den, rätta kvarvarande fel och försöka

igen. Om filen är korrekt noterar datorn hur lång tid det tog att rätta felen och sorterar in resultatet i en hiscore-lista.

Det är viss slumpmässighet i genereringen och därmed en viss variation i längd och antal fel mellan olika filer, så det är tillåtet att upprepa det hela tills man får en tid man är nöjd med. I hiscore-listan bevaras åtminstone den bästa resultatet och det senaste resultatet från varje deltagare. Man kan titta på hiscore-listan med kommandot

```
textred hiscore
```

Programmet `textred` finns i katalogen `~hallgren/Intro/`, så det är lämpligt att göra

```
set path = ( $path ~hallgren/Intro/ )
```

innan man startar, annars måste man skriva `~hallgren/Intro/textred ...` varje gång man använder kommandot. (Jämför Uppgift 2 i Laboration nr 6.)

Innan man börjar kan det vara lämpligt att läsa i Gula Boken eller annan dokumentation om vilka kommandon det finns i textredigeringsprogrammet som kan snabba upp arbetet. Oftast går det fortare att utföra kommandon med tangentbordet än att göra det genom att välja kommandon från menyer med musen. Kan man med en knapptryckning ta bort ett helt ord eller en hel rad? Kan man markera ett textavsnitt, klippa ut det och klistra in det på ett annat ställe, utan att använda musen? Hur göra man för att söka efter en viss teckenkombination? Kan man automatiskt ersätta alla förekomster av en viss teckenkombination med något annat?

- Varsågods att börja! För att bli godkänd på denna uppgift måste man vara med på hiscore-listan. Det är tillåtet att fortsätta försöka förbättra sin placering på hiscore-listan utanför laborationstid. Man kan även tävla från sitt personliga konto, om man vill få med sitt eget namn i hiscore-listan.

### Uppgift 7. Workingday

Detta är en uppgift från Victor Kouzmine, k97kovi@chestud.chalmers.se.

Denna uppgift går ut på att registrera alla E1-studenter i "Workingday"s databas, samt repetera Internet-kunskaper. "Workingday" är ett nystartat projekt av några studenter från bl a Chalmers, Handelshögskolan i Stockholm, London School of Economics, och Princeton University. Projektet handlar om att samla in studenters meritförteckningar via en webbsida och erbjuda olika svenska och europeiska företag möjligheten att marknadsföra sig mot dessa studenter i rekryteringssyfte. Även för dig som nyliken har börjat på Chalmers, är det högst viktigt att du registrerar dig i Workingdays databas, eftersom många företag kommer att skicka ut olika erbjudanden om extrajobb, sommarjobb, examensarbete. Med tanke på att det är svårt för studenter att hitta praktikplatser (17 veckors praktik är obligatoriskt för att erhålla civilingenjörsexamen) och sommarjobb har alla studenter mycket att vinna på att registrera sig!

- Öppna Netscape och gå till adressen [www.workingday.com](http://www.workingday.com). Klicka sedan på **Register CV**, som kommer precis under rubriken **STUDENTS**. Om du inte lyckas klicka sig fram till registreringsidan, kan du gå dit direkt genom att gå till adressen [http://www.workingday.com/register/login\\_regcv\\_new.html](http://www.workingday.com/register/login_regcv_new.html). Läs texten och klicka sedan på **Next**

och fyll i blanketten som dyker upp. När du är klar med första sidan klickar du på **Next** längst ner till höger och gör så, tills du kommer fram till första sidan igen. Då väljer du ett loginnamn och lösenord. Spara dessa, för du måste gå in och uppdatera dina uppgifter minst varannan månad för att vara kvar i databasen.

Båda personerna i labgruppen måste registreras var för sig. Efter du har registrerat sig kan du även få en ergonomisk musmatta gratis om du skickar mail med din hemadress till [info@workingday.com](mailto:info@workingday.com).

### 3 Avslutande kommentarer

Motiveringen till de flesta uppgifterna i denna laboration var som sagt att ge lite praktiskt erfarenhet av saker som kan komma till nytta i senare kurser. För att få rutin i filhantering och textredigering krävs det antagligen mer än ett tvåtimmarspass med några få övningar, så fortsatt gärna att öva på egen hand, för det kan ju vara skönt att slippa fastna och slösa tid på sådana detaljer i senare kurser, t ex E2 Programmeringsteknik. Denna kurs beskrivs förresten på webbsidan

<http://www.cs.chalmers.se/Cs/Grundutb/Kurser/e2pt/>

Programmen *WinZip* och *gzip* har egna webbplatser:

[www.winzip.org](http://www.winzip.org)

[www.gzip.org](http://www.gzip.org)

[www.workingday.com](http://www.workingday.com) är förstås inte det enda sättet att hitta praktikjobb och examensarbeten. Andra användbara länkar kan man hitta t ex på följande webbsidor:

<http://www.etek.chalmers.se/lankar.html>

<http://www.edkan.chalmers.se/edgrund.html>

# Laborationer i kursmomentet Datoranvändning E1

<http://www.etek.chalmers.se/~hallgren/Eda/>

## Laboration nr 10: Att göra webbsidor

1996, 1997 Magnus Bondesson  
1998 och 99-11-28 Thomas Hallgren

**Sista godkännandedag: torsdag 99-12-09 kl 17.00**

### 1 Introduktion


Denna laboration handlar om diverse saker som man kan ha nytta av när man skapar webbsidor. Det ingår också en övning på att hämta hem ett program från Internet, kompilera det och använda det. Det blir också lite repetition av saker som dykt upp tidigare i kursen.

Kommandon och program som dyker upp i denna laboration är: `gnnpres`, `xpaint`, `xv`, `giftrans`, `vrweb`, `tar`, `make`, `whirlgif` och `xanim`.

#### 1.1 Förberedelser

Skumma igenom häftet. I vissa uppgifter finns det mycket förklarande text att läsa. Har du inte redan stor erfarenhet av att skapa webbsidor är det lämpligt att gå på föreläsningen.

#### 1.2 Redovisning

Denna laboration redovisas genom att man skapar en webbsida som innehåller svaren från uppgifterna. Saker som ska tas med i redovisningen är markerade med  .

För att bli godkänd på denna laboration ska du visa upp redovisningssidan för en handledare.

#### 1.3 HTML och verktyg för att skapa HTML-dokument

Webbsidor skrivs i beskrivningsspråket HTML (HyperText Markup Language). Många ordbehandlingsprogram (men tyvärr inte den version av FrameMaker som finns på E) kan nuförtiden spara dokumenten i HTML-format.

HTML är dock ett ganska enkelt format, så man kan också använda vanliga textredigeringsprogram, t ex `nedit`. En ganska lättläst HTML-guide finns på

<http://www.htmlhelp.com/reference/wilbur/>

och man kan också lära sig mycket på att titta på HTML-koden för existerande webbsidor, t ex genom att använda menyalternativet **View/Page Source** i *Netscape*. När man

hittar något bra på en webbsida som man inte riktigt förstår kan man gå till HTML-guiden och slå upp det.

Det finns också program speciellt gjorda för att skapa webbsidor. *Netscape Communicator* har *Composer* inbyggt. Ett annat sådant som finns på E är *GNNpress*.

I denna laboration kan du i de flesta uppgifterna välja om du vill använda *Netscape Composer*, *GNNPress* eller något annat verktyg för att skapa webbsidor, eller om du vill skriva HTML-koden själv i ett textredigeringsprogram.

*Netscape Composer* kan startas på flera sätt: från menyn **Communicator/Page Composer**, **File/Edit Page**, eller **File/Open Page** och sedan knappen **Open in Composer**.

*GNNpress* startas med kommandot `gnnpress`. Det används på ungefär på samma sätt som vanliga ordbehandlingsprogram. Men allt är ändå inte självklart, så fråga handledare eller använd den **Tutorial** som finns under **Help**. Bläddra igenom menyerna för att få en allmän uppfattning om programmet.

## 2 Uppgifter

### Uppgift 1. e1person


Det blev lite problem med `e1person` förra veckan, så om du inte lyckades registrera dina personuppgifter då, gör ett nytt försök. Om du får ett felmeddelande som innehåller "cannot open display" hjälper det att göra

```
xhost +
e1person
xhost -
```

### Uppgift 2. Skapa en katalog för denna laboration

I de följande övningarna kommer vi att skapa ett antal filer. För att samla dem på ett ställe och hålla isär dem från alla andra filer börjar vi med att skapa en ny katalog. Kalla den `Lab10`. (När senare uppgifter refererar till `Lab10` är det denna katalog som menas.)

### Uppgift 3. Gör i ordning början på redovisningssidan

 I den gamla vanliga katalogen, `~hallgren/Intro/`, finns filen `labredovisning.html`. Kopiera den till katalogen `Lab10` och öppna den i *NEdit*. På några ställen står det `XX`. Byt ut dessa mot lämpliga uppgifter.

*NEdit* har en finess för att göra vissa kända typer av filer, t ex HTML-filer, mer lättlästa. Genom att välja **Preferences/Highlight Syntax** färgsätts HTML-markeringarna så att de blir lättare att själva från den vanliga texten. Man kan få denna, och andra inställningar, att aktiveras automatiskt när man startar *NEdit*, genom att göra de inställningar man vill ha i menyn **Preferences/Default Settings** och sedan välja **Preferences/Save Defaults**.

Kontrollera med *Netscape* att det ser bra ut, och jämför webbsidans utseende med HTML-koden. Lägg märke till HTML-filens uppbyggnad:

- HTML-koden består av ett antal element. De flesta element har en startmarkering, t ex `<H2>`, ett innehåll, och en matchande slutmarkering, t ex `</H2>`. Elementens namn kan skrivas med stora eller små bokstäver, så t ex `<h1>` och `<H1>` betyder samma sak.
- Innehållet i ett element kan vara en blandning av vanlig text och (mindre) element.
- För vissa element är det tillåtet att utelämna slutmarkeringen. Vissa element har inget innehåll och då ska slutmarkeringen alltid utelämnas. För vissa obligatoriska element kan både startmarkeringen och slutmarkering utelämnas.
- För vissa element kan *attribut* (extra information) anges i startmarkeringen. Två exempel är bilder och länkar:

```
<IMG SRC="url" >
```

```
<A HREF="url" >länktext</A>
```

IMG är exempel på ett element utan innehåll och slutmarkering.

- Ett komplett HTML-dokument innehåller alltid exakt ett HTML-element, som i sin tur alltid innehåller ett HEAD-element och ett BODY-element. HEAD-elementet ska innehålla ett TITLE-element. BODY-elementet innehåller det som visas i webbläsarens fönster.
- BODY-elementet utgör alltså den största delen av dokumentet. Här finns det en mängd olika element att använda för att beskriva dokumentets struktur, H1, H2, ... H6 ger rubriker på olika nivåer, A ger länkar, IMG ger bilder, UL och OL ger onummerade respektive nummerade listor, P börjar på nytt stycke, mm. På sidan <http://www.htmlhelp.com/reference/wilbur/overview.html> finns en översikt över alla HTML-element. (Det är HTML 3.2 som beskrivs. Det finns nyare versioner av HTML. Olika webbläsare har också mängder av icke standardiserade utökningar, som tyvärr används alltför flitigt.)

#### Uppgift 4. Låna en bild

Bilder från WWW-sidor kan man plocka ut och spara för egen användning. Detta får man dock bara göra om man vet att bilden är fri. Framför allt skall man akta sig för att ta bilder från företagssidor, inklusive bl a tidningssidor.

Man kan göra på två sätt:

1. Kopiera bilden och länka till den egna kopian. I Netscape gör man enklast så här: Placera markören på bilden. Tryck på musens högra knapp och välj **Save Image As...** i den meny som kommer fram. Spara bilden i lämplig katalog med lämpligt namn.
2. Låta bli att kopiera bilden och länka till originalet. Man kan kopiera adressen till bilden genom att välja **Copy Image Location** menyn som fås med högra knappen, och sedan klistra in den genom att klicka med mittenknappen där man vill ha den. (Exakt hur man klistrar in beror på vilket program man klistrar in i.)

För att infoga bilden i HTML-dokumentet skriver man

```

```

om man redigerar HTML-koden själv i ett textredigeringsprogram. I *Netscape Composer* väljer man **Insert/Image...** från menyn. i *GNNPress* väljer man **Element/Image** i menyn. Istället för *url* ovan sätter man in bildens adress. Om man valde att kopiera bilden till den katalog där HTML-dokumentet finns räcker det med filnamnet. Om man länkar till en bild är det här man kan klistra in genom att klicka med mittenknappen, om man har tidigare har gjort **Copy Image Location**.

☞ På [www.iconbazaar.com](http://www.iconbazaar.com) finns det en stor samling bilder av olika slag som det är tillåtet att kopiera ifrån. Leta rätt på någon kul bild och stoppa in den i redovisningssidan. Skriv också en liten kommentar om var bilden kommer ifrån och vilka villkor som gäller för användandet av dessa bilder.

### Uppgift 5. Färgtabellen

De flesta datorerna på E har ett system för färggrafik som gör att man inte kan ha mer än 256 olika färger samtidigt på skärmen. (Bildminnet lagrar bara en byte per pixel.) Vilka färger som kan visas bestäms av en tabell, den s k paletten, vars innehåll kan förändras. Även persondatorer som Mac och PC brukar ha detta system, men de senaste åren har grafikkort och minnen blivit så billiga att det på de flesta moderna datorer brukar på vara möjligt att ställa om till ett system utan begränsningar på antalet färger. (Bildminnet lagrar för varje pixel 3 bytes som anger vilken färg som ska visas, istället för en byte som är ett index till paletten.)

Titta på färgtabellen med kommandot `xcmmap`. Förmodligen ser du en ganska slumpmässig blandning av olika färger. Råkar du sitta vid en av datorerna med 24-bitarsgrafik visar `xcmmap` antagligen bara nyanser av en och samma färg.

☞ Gör en skärmavbildning av färgtabellen och ta med i redovisningsfilen. Du har tidigare gjort skärmavbildningar med `FrameMaker`, men prova att göra det med `xv` istället genom att

- starta `xv`,
- klicka med höger musknapp i `xv`:s fönster för att få fram en kontrollpanel,
- använda **Grab**-knappen och följa instruktionerna som ges för att göra skärmavbildningen
- och sedan **Save**-knappen för att spara bilden i katalogen `Lab10`. Välj ett lämpligt filformat, dvs GIF eller JPEG.

Bilden kan sedan infogas i HTML-dokumentet på samma sätt som i förra uppgiften.

### Uppgift 6. Bildbehandling med `xv`

- Öppna filen `~hallgren/Intro/bild1.gif` med `xv`. Spara en kopia av den i katalogen `Lab10`. Experimentera sedan med bildbehandlingsoperationerna under **Algorithms**. Du kan ångra med **Undo All**.

- Öppna filen `~hallgren/Intro/bakelse.jpg`. Som du ser är bilden för mörk och bakelsen ser inte så aptitlig ut. Det vanligaste problemet när bilder är för ljusa eller för mörka är att det s k gamma-värdet inte stämmer. Lagg in en gammakorrigering (omkring 1.4 är antagligen lagom), halvera bildens storlek och spara med ett nytt namn i katalogen Lab10. (Tips: använd **Windows/Color Editor** i kontrollpanelen.)

- ☞ • Lagg till de båda bilderna i redovisningsfilen.

Kommandot `man xv` ger ett närmast innehållslöst manualblad, som dock leder dig till en mycket omfattande manual i PostScript (adressen är fel, skall fn vara `/usr/local/pd/inst/xv-3.10a/lib/xvdocs.ps`).

Man kan läsa om gammavärden på

<http://www.vtiscan.com/~rwb/gamma.html>

### Uppgift 7. Ändra bakgrundsbild

Som du säkert har sett om du har surfat på webben, så kan man ha bakgrundsbilder på webbsidor. Detta gör man genom att stoppa in adressen till bilden i attributet `BACKGROUND` i dokumentets `BODY`-elementet, dvs man skriver

```
<BODY background="...">
```

där `...` byts ut mot bildens adress. I *GNNPress* kan man lägga in bakgrundsbilder med menyalternativet **Format/Page Attributes...** I *Netscape Composer* kan man använda **Format/Page Colors and Properties**.

- ☞ Leta upp en bakgrundsbild på [www.iconbazaar.com](http://www.iconbazaar.com) (eller på något annat ställe) och lägg in den i redovisningsdokumentet. Välj en *ljus* bild med ett *dämpat* mönster, så att texten fortfarande är lättläst.

### Uppgift 8. Bilder med transparent bakgrund

Du har förmodligen noterat att i andras HTML-dokument finns det bilder liknande den mellersta nedan, medan du själv bara lyckats åstadkomma typen till vänster. Skillnaden är att bakgrunden lyser igenom allt som är vitt. GIF-formatet tillåter oss att lagra information om vilken färg som skall vara transparent. I den mellersta bilden är vitt transparent, medan i den högra grönt är det.



Programmet `giftrans` (som finns på det vanliga stället) kan användas för att göra transparent bakgrund. Det används så här:

<pre>giftrans -l E.gif</pre>	Visar färgtabellen i GIF-bilden <code>E.gif</code> .
------------------------------	--



<code>giftrans -t index E.gif &gt;ETRANS.gif</code>	Gör en kopia av bilden med <i>index</i> (0-255) som transparent färg.
<code>giftrans</code>	Ger information om andra möjligheter.

☞ Gör en transparent kopia av en bild du ritat själv med `xpaint` eller av `~hallgren/Intro/E.gif`. Prova med *Netscape*. Lägg in både den ursprungliga bilden och den transparenta i redovisningsfilen.

### Uppgift 9. Animerad GIF

Titta på sidan `file:/u1/ext/hallgren/Intro/animerad_gif.html` med *Netscape*. Den typ av bilder som visas kallas animerade GIF-bilder. Filformatet GIF tillåter att flera bilder lagras i en fil.

### Uppgift 10. Hämta program via Internet

Om man inte har ett program för lösa ett visst problem, t ex skapa animerade GIF-bilder, kan man i många fall hitta ett på Internet. När det gäller Mac/PC är programmen man hämtar ofta klara att köras. När det gäller UNIX-program är det av olika anledningar vanligt att man får källkoden till programmet. Man måste då kompilera det själv. Ofta är programmen skrivna i C, eftersom en C-kompilator ingår som standard i de flesta UNIX-system.

Det finns ett program som heter *WhirlGIF* som tillverkar animerade GIF-bilder. Det finns på webbsidan

`http://www.msg.net/utility/whirlgif/`

Hämta hem det (filen heter `whirlgif.tar.gz`) och spara det i katalogen `Lab10`. Filnamnsslutet `.tar.gz` antyder att det är ett tar-arkiv som komprimerats med `gzip`.

Packa upp filen. (Du kommer väl ihåg från Laboration nr 9 hur man gör?) En ny katalog som heter `whirlgif` skapas. Flytta dig in i den. Se efter vilka filer som finns där. Du kan kompilera källkoden genom att bara skriva

```
make
```

Då skapas ett körbart program `whirlgif`, med vilket animerade GIF-bilder kan bildas. Titta hastigt på `whirlgif.doc` som beskriver hur man går tillväga.

### Uppgift 11. Tillverka animerad GIF-sekvens

Stanna kvar i katalogen ovan. Skapa 2-3 GIF-bilder med `xpaint` (alternativt med *Maple* eller *MATLAB*) som skall bilda underlag för animeringen. Kalla dem t ex `G1.gif`, `G2.gif` och `G3.gif`. Skriv sedan

```
whirlgif G*.gif > ANIMG.gif
```

så bildas den animerade filen. (Notera att vi kunde använda filnamnsmönstret `G*.gif` istället för att räkna upp namnen på alla bilderna tack vare att vi gav dem liknande namn. Notera också att vi använder omdirigering för att spara resultatet i en fil.)

- ☞ Lägg in den animerade bilden i redovisningsdokumentet och kontrollera att animationen fungerar i *Netscape*.

### Uppgift 12. Klickbara kartor med GNNPress

Gör ett nytt HTML-dokument med *GNNPress* genom att välja **File/New Page** i menyn. Fyll i filnamn och titel. Välj därefter **Image** och låt den utgöras av [http://www.chalmers.se/GIF/CTH\\_karta.gif](http://www.chalmers.se/GIF/CTH_karta.gif). Vi skall nu göra så att när man klickar på kartan i närheten av E så skall man komma till E-s hemsida <http://www.etek.chalmers.se> och när man klickar nära Matematiskt centrum skall man komma till <http://www.cs.chalmers.se>. Markera kartan genom att dubbelklicka. Välj **Element/Image map...** från menyn. Det kommer upp ett nytt fönster. Välj något av de tre verktygen rektangel, cirkel och polygon. Drag kring E. Fyll i adressen i rutan **Location**. Drag på samma sätt kring Matematiskt Centrum och fyll i adressen. Spara med **SAVE**. Avmarkera bilden. För markören över bilden. Klicka nära något av de två ställena. Du märker förmodligen nu om inte tidigare att *GNNPress* inte är sämre än *Netscape* på vanlig webbläsning.

- ☞ Spara ditt dokument i katalogen Lab10. Gör en länk till det från redovisningsdokumentet. Prova att länken och den klickbara kartan fungerar i *Netscape*.

### Uppgift 13. Tillverka en logotyp

Det finns ett bildbehandlingsprogram för UNIX som heter GIMP (se [www.gimp.org](http://www.gimp.org)). Det är gratis och liknar Adobe PhotoShop (som inte är gratis). GIMP är programmerbart, så folk har gjort diverse små program för att tillverka tjugusiga logotyper, knappar, animationer och annat kul. GIMP finns inte på E, men det finns en webbsida, [www.cooltext.com](http://www.cooltext.com), där man kan tillverka egna logotyper genom att fjärrstyra GIMP.

- ☞ Tillverka en logotyp och lägg in den överst på redovisningssidan.

### Uppgift 14. VRML

WWW har gett upphov till en mängd idéer. En sådan är att man skall kunna hämta 3D-modeller från webbservern (precis som bl a vanliga HTML-sidor) och sedan interaktivt kunna vandra runt i dessa världar. Det finns en framväxande standard för sådana modeller i form av ett språk kallat VRML (Virtual Reality Modelling Language). Starta programmet `vrweb` och öppna dokumentet `~hallgren/Intro/BANAN.wrl`. Utforska menyerna och experimentera. Öppna sedan det större dokumentet `~hallgren/Intro/GRAZ.wrl` och lek litet. Titta gärna även på det första dokumentet med `tex NEdit` eller lista det bara i fönstret med `cat`.

**Anm.** VRML-dokument kan också öppnas från *Netscape* om man har rätt inställningar.

## 3 Frivilliga extrauppgifter

### Uppgift 15. Interaktiva webbsidor och JavaScript

Animerade GIF-bilder kan göra webbsidor mer levande, men betraktaren kan ändå inte göra så mycket mer än att titta på sidan och följa länkar.

Man kan skapa webbsidor med lite mer interaktiva inslag genom att utöka HTML-dokumentet med ett avsnitt med programkod i JavaScript. Här visas ett enkelt exempel, där användaren matar in ett uttryck, trycker på knappen **Räkna ut** och får resultatet.



JavaScript-koden, som finns i filen `~hallgren/Intro/javascript.html`, är i detta exempel uppdelad i två delar. Den första delen är en hjälpfunktion som ligger i dokumentets HEAD-del:

```
<SCRIPT LANGUAGE="JavaScript">
function compute(form) {
    if (confirm("Är det verkligen detta som du vill ha
beräknat?"))
        form.result.value = eval(form.expr.value)
    else
        alert("Det var väl kul. Försök gärna igen!")
}
</SCRIPT>
```

Den andra delen finns i BODY-delen och skapar de synliga delarna och använder hjälpfunktionen:

```
<FORM>
Skriv ett uttryck:
<INPUT TYPE="text" NAME="expr" SIZE=15 >
<INPUT TYPE="button" VALUE="Räkna ut"
ONCLICK="compute(this.form)" >
<BR>
Resultat:
<INPUT TYPE="text" NAME="result" SIZE=15 >
</FORM>
```

☞ Kopiera in detta JavaScript-exempel i ditt redovisningsdokument. Detta görs lämpligen med `nedit`. Om du har redigerat din webbsida med *GNNPress* innan är det bäst att spara och stänga dokumentet i först.

### Uppgift 16. Applets

Applets är ett annat sätt att göra interaktiva inslag i webbsidor som först dök upp i Suns webbläsare *HotJava*. Applets är småprogram som vanligen skrivs i språket Java, som är så populärt just nu. Gå till

<http://www.javasoft.com/applets/>

och titta t ex på appletterna *Pythagoras* och *Nuclear Plant* under rubriken *Applet Archive*.

☞ Lägg också märke till att det finns instruktioner om hur man lägger till en applet till en webbsida. Leta upp en applet som du tycker om och kopiera den till redovisningssidan!

## 4 Avslutande kommentarer

Du som är intresserad av fler detaljer kring det som tas upp här hänvisas till det rikliga bokutbudet eller kanske hellre till det ofta mycket aktuella material som finns på nätet.

### 4.1 Information på webben

Redan i anslutning till de olika uppgifterna har det dykt upp ett antal webbadresser. Här följer ytterligare några.

På [www.htmlhelp.com](http://www.htmlhelp.com) kan man hitta diverse nyttig information för webbsidemakare, bland annat kortfattade och överskådliga beskrivningar av HTML.

När man bläddrar i svenska Yahoo ([www.yahoo.se](http://www.yahoo.se)) hittar man en del HTML-kurser, nybörjarguider och annat på svenska. Två exempel är:

<http://www.april.se/htmlkurs/>

<http://www.skolverket.se/skolnet/htmlkurs/>

Om man söker i Yahoo eller andra webbkataloger hittar man ganska många webbsidor med samlingar av symboler, linjer, prickar, animerade GIF-bilder och annat som man kan utsmycka sina webbsidor med. Vi har redan sett [www.iconbazaar.com](http://www.iconbazaar.com). Ett annat exempel är [www.mediabuilder.com](http://www.mediabuilder.com).

Genom att söka efter JavaScript i Yahoo hittar man lätt många JavaScript-exempel.

När man gör webbsidor testas man förstås att de ser bra ut i den egna webbläsaren, men olika webbläsare hanterar felaktig HTML-kod på olika bra, så det kan vara bra att kontrollera att HTML-koden följer standarden. Det kan man göra på webbsidan [validator.w3.org](http://validator.w3.org). Det finns också ett program som heter *weblint* som kontrollerar HTML-filer, men det verkar inte vara installerat på E.

På [www.useit.com](http://www.useit.com) skriver Jakob Nielsen om hur man gör bra webbsidor, vad som är bra och dåligt på webben, m m. I hans "AlertBox" hittar man t ex artiklarna *Why Yahoo is good*, *Top ten mistakes of web design*, *Why frames suck*, m m. Exempel på en svensk webbplats där webbdesignfrågor diskuteras är

<http://www.idg.se/webstudio/>