Laborationer i kursmomentet Datoranvändning E1

`http://www.etek.chalmers.se/~hallgren/Eda/`

Laboration nr 9: Mer om WWW och HTML

1996, 1997 Magnus Bondesson
98-11-15 Thomas Hallgren

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: `gnnpress`, `xpaint`, `xv`, `giftrans`, `vrweb`, `tar`, `make`, `whirlgif` och `xanim`.

1.1 Förberedelser

Det kan vara lämpligt att titta igenom gamla laborationer, speciellt med tanke på Uppgift 3.

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

Som vi såg redan i Laboration nr 1 så skrivs webbsidor i beskrivningsspråket HTML. 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/Document 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. Ett sådant är *GNNpress* och det finns på E.

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

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

Det är ganska många uppgifter i denna laboration. Om ni inte hinner klart med allt på två timmar går det bra att fortsätta nästa vecka.

Uppgift 1. 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 `Lab9` eller något liknande. (När senare uppgifter refererar till `Lab9` är det denna katalog som menas.)

Uppgift 2. Gör i ordning början på redovisningssidan

☛ I den gamla vanliga katalogen, `~hallgren/Intro/`, finns filen `labredovisning.html`. Kopiera den till katalogen `Lab9`. På några ställen står det XX. Byt ut dessa mot lämpliga uppgifter.

Kontrollera med Netscape att det ser bra ut.

Uppgift 3. En lathund över UNIX-kommandon

I laborationerna har vi nu stött på omkring 40 olika UNIX-kommadon. Det kanske är lite jobbigt att komma ihåg utantill hur alla fungerar. En liten lathund som sammanfattar kommandona kan vara till stor hjälp.

☛ Skapa en lathund över kommandona som använts. Den ska innehålla en enradsbeskrivning av varje kommando. Det finns en länk till en påbörjad lathund på kursens hemsida. Kopiera den till katalogen `Lab9` och utgå från den. Gör en länk från redovisningssidan till lathunden. Kontrollera att länken fungerar.

Det är tillåtet att samarbeta med en annan labbgrupp.

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 få adressen till bilden genom att välja **Copy Image Location** menyn som fås med högra knappen.

☛ På `www.iconbazaar.com` finns det bilder som det är tillåtet att kopiera. 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 anvädadet 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.) Palettens innehåll kan dock ändras. Även persondatorer som Mac och PC brukar ha detta system, men det brukar på dem numera också vara möjligt att ställa om till ett annat system som innebär att man kan ha många fler färger samtidigt.

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

☛ Gör en skärmavbildning av färgtabellen och ta med i redovisningsfilen.

Uppgift 6. Bildbehandling med xv

- Öppna filen `~hallgren/Intro/bild1.gif`. Spara den sedan i katalogen `Lab9`. 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. Lägg in en gammakorrigering (omkring 1.4 är antagligen lagom), halvera bildens storlek och spara i katalogen `Lab9`.
- ☛ Lägg 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 f n 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 backgrundsbild

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...**

Leta upp en bakgrundsbild på `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:

`giftrans -l E.gif`	Visar färgtabellen i GIF-bilden `E.gif`.
`giftrans -t` *`index`* `E.gif >ETRANS.gif`	Gör en kopia av bilden med *index* (0-255) som transparent färg.
`giftrans`	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 transparanta 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. Lagringsformatet 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 `Lab9`. Filnamnsslutet `.tar.gz` antyder att det är ett tar-arkiv som komprimerats med `gzip`. Programmet `tar` (tape archive) utformades från början för att göra backup av filer (vanligen hela kataloger med filer och allt) på band, men används även när man vill packa ihop många filer till ett paket. (Se `man tar` för uttömmande information.)

Packa upp filen med kommandona

```
gunzip whirlgif.tar.gz
tar xvf whirlgif.tar
```

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 (omdirigering!)

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

så bildas den animerade filen. (`G*.gif` är ett mönster som ersätts med passande filnamn av kommandotolken innan kommandot utförs. Här får vi alla filer vars namn börjar på *G* och slutar på `.gif`.)

☛ Lägg in den animerade bilden i redovisningsdokumentet och kontrollera att animeringen fungerar i *Netscape*.

Uppgift 12. Klickbara kartor med GNNPress

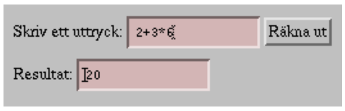
Gör ett nytt HTML-dokument med *GNNPress* genom att välja **File/New Page** i menyn. Fyll i filnamn och titel. Välj därefter **Image** och låt den utgöras av `http://www.chalmers.se/GIF/CTH-karta-WWW.GIF`. Vi skall nu göra så att när man klickar på kartan i närheten av E så skall man komma till E-s hemsida `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 webbläddring.

☛ Spara ditt dokument i katalogen `Lab9`. Gör en länk till det från redovisningsdokumentet. Prova att länken och den klickbara kartan fungerar i *Netscape*.

Uppgift 13. Interaktiva webbsidor och JavaScript

Animerade GIF-bilder kan göra webbsidor mer levande, men betraktaren kan ända 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 kod 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 dokumentents 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 *GNNPress* först.

Uppgift 14. 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!

Uppgift 15. Tillverka en logotyp

Det finns ett bildbehandlingsprogram för UNIX som heter GIMP (se `www.gimp.org`). Det är gratis och liknar programmet PhotoShop (som inte är gratis). GIMP är programmerbart och folk har gjort diverse små program för att tillverka tjusiga logotyper, knappar, animeringar och annat kul. GIMP finns inte på E, men det finns en webbsida, `www.cooltext.com`, där man kan tillverka egna logotyper genom att fjärrstyra GIMP.

☛ Tillverka en logotyp och lägg in den överst på redovisningssidan.

Uppgift 16. 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 t ex *NEdit* eller lista det bara i fönstret med `cat`.

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

Uppgift 17. Multimedia: xanim

X-programmet `xanim` kan visa animerade GIF-bilder och en mängd andra format för rörliga bilder, bl a MPEG, AVI (från Microsoft) och QuickTime (från Apple), som även kan innehålla ljud. Prova denna aktuella trailer (som jag hittade via The Internet Movie Database på `www.imdb.com`):

```
xanim +Sr ~hallgren/Intro/trumanhi.mov
```

Du kan stoppa/starta bildväxlingarna med mellersta musknappen. Vänstra/högra stegar en bild bakåt/framåt. Flaggan `+Sr` gör att man kan ändra storlek på fönstret medan filmen spelas. Programmet kan stoppas med `q` i filmfönstret.

3 Avslutande kommentarer

Du som är mycket 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.

3.1 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` 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`) 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`. Ett annat exempel är `www.mediabuilder.com`.

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

När man gör webbsidor testar 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 följder standarden. Det kan man göra på webbsidan `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` 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/klubben/webstudio/`