

DD1310 Programmeringsteknik för CMAST2-ITSY och Open1

Laborationer läsåret 2017/2018

Fyll i ditt namn och personnummer med bläck eller motsvarande. OBS: Om inte denna kvittenssida tas med vid redovisningen får du ingen kvittens (resultatet rapporteras dock in i rapporteringssystemet, Rapp, på CSC).

Kursledare och examinator är Vahid Mosavat, vahid@kth.se.

Namn Personnr

Laborationer

Laboration 1	Godkänt den (bonusdatum 2017-09-01)	Kvitteras	Timmar
Laboration 2	Godkänt den (bonusdatum 2017-09-07)	Kvitteras	Timmar
Laboration 3	Godkänt den (bonusdatum 2017-09-14)	Kvitteras	Timmar
Laboration 4	Godkänt den (bonusdatum 2017-09-22)	Kvitteras	Timmar
Laboration 5	Godkänt den (bonusdatum 2017-10-11)	Kvitteras	Timmar

P-del

Spec	Godkänt den	Kvitteras	
Redovisning	Godkänt den	Kvitteras	Timmar
Doctest	Godkänt den	Kvitteras	
Granskning	Godkänt den	Kvitteras	

Laboration 1: Introduktion till datormiljön och Python

Nyckelord: *katalog, katalogträd, rot, fil, källkod, program, exekvering och tilldelning*

Mål: *Att du efter laborationen ska klara dig på egen hand i datorsalarna, ha registrerat dig på kursen och kan beräkna uppgifter i python.*

Föreberedelse: *Skaffa användarnamn och lösenord och hitta en jämnbra labbkompis*

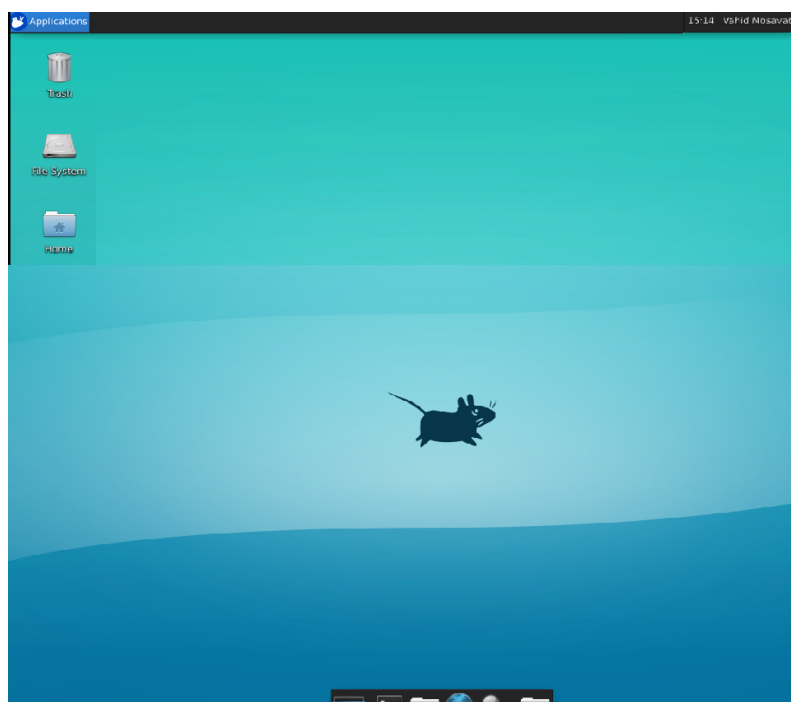
Sammanfattning av labben

I den här labben ska du lära dig om operativsystemet **LINUX** och redigeringsprogrammet **Emacs**. Du ska också skriva enkla Pythonprogram och registrera dig på kursen. Efter den här laborationen ska du klara dig på egen hand i datorsalarna. Betydligt mer information om hur man hanterar datorerna finns i unixhäftet som säljes på CSCs studentexpedition.

Inloggning

Hitta en labbpartner, så att ni kan bilda en grupp av 2 personer. Gruppen får absolut inte vara mer än 2. Sätt dig vid någon dator i någon av salarna gul, grön eller brun på plan 4 i E-huset, CSC skolan.

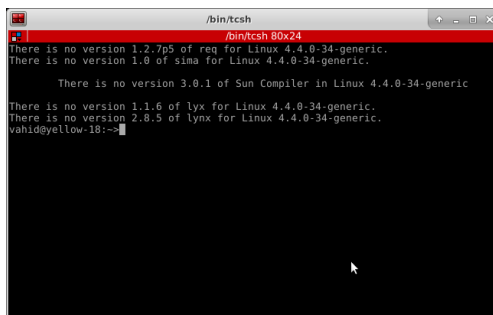
Logga in nu med de inloggningsuppgifter du har fått. Efter inloggningen kommer skrivbordet visas. Skärmen kommer ungefär se ut så här (bilden kanske inte ser ut exakt likadan med det visas på skärmen men det spelar inte så stor roll):



På vänstra delen av skrivbordet finns frontpanelen. Den används för att starta vissa program, ändra inställningar m.m. Klicka på första ikonen uppfifrån på frontpanelen. Då dyker upp ett sökfält. Om du skriver "Terminal" i sökfältet visas terminal ikonen bl.a ikoner. Klicka på terminal-ikonen.



Ett nytt fönster, terminalfönster, öppnas. Terminalfönstret motsvarar ett DOS-fönster på en PC och ser ut som nedan.



När du skriver i det fönstret ger du kommandon till kommandotolken. Kom ihåg att kommandotolken skiljer mellan stora och små bokstäver.

Webbläsare och kursinformation

Programmet *firefox* är en webbläsare som startas genom att klicka på **Application** och välj sedan **Web Browser** från listan. Alternativt kan du använda kommandot **firefox &** i ett terminalfönster.

Skriv nu följande URL-adress i adressfältet längst upp på webbläsarens fönster:

<https://www.kth.se/social/course/DD1310/>

Du ska nu få upp en sida med kursinformation för programmeringsteknikkursen för CMAST2-ITSY och Open1. Denna sida är viktig! Här kommer vi att lägga ut information under kursens gång, så vänj dig att alltid titta på sidan när du loggar in. Använd dina inloggningsuppgifter för att logga in på sidan (om du hittar en knapp med texten "logga in" på sidan), då kommer du att få tillgång till all material på kursen.

För att slippa klicka dig fram till sidan varje gång ska du spara adressen till sidan med ett bokmärke. Du gör det genom att välja menykommandot

Bookmarks → Bookmark This Page.

Registrering på kursen

Nu när du provat webbläsaren firefox är det dags att du registrerar dig på kursen. Det gör du genom att ange adressen `http://rapp.csc.kth.se` i webbläsarens adressfält, logga in med KTH.SE inloggningsuppgifter. Sedan är det bara följa instruktionerna.

Tillkalla handledare med Queue

Om du behöver hjälp med något eller vill redovisa använder du webbsidan `queue.csc.kth.se`. För att använda Queue krävs att logga in med KTH.SE inloggningsuppgifter, se därför till att du har inloggningsuppgifterna tillhands. Möjlighet att få hjälp eller redovisa finns endast under schemalagda labbtillfällen.

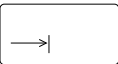
Terminalfönstret

Terminalfönstret har en mängd kommandon och nedan finns exempel på de mest användbara för denna kurs. Se tabellen på nästa sida.

Vissa kommandon skriver man i terminalfönstret (allt ovanstående går också att göra med skrivna kommandon). Till exempel kan du lista alla filer i en katalog med kommandot `ls` (list). Vill du lista alla underkataloger och filer i kurskatalogen skriver du `ls /info/prgt/exempelfiler`. Vill du skapa en ny katalog med namnet `prgo17` skriver du `mkdir prgo17` i terminalfönstret.

Det du framför allt behöver veta är hur du flyttar dig mellan olika kataloger i terminalfönstret. För att gå ner i katalogen `prgo17` skriver du `cd prgo17`, vilket utläses *change directory to prgo17*. För att gå upp en nivå skriver du `cd ..` (glöm inte mellanslaget mellan `cd` och de två punkterna).

Det finns genvägar för att slippa skriva så mycket i terminalfönstret; till exempel behöver du sällan skriva ut hela filnamnet. Tryck på TAB tangenten

 (längs till vänster) så fylls kommandon, filnamn och katalognamn i om de är entydiga (detta kallas "TAB completion" på engelska). Vill du få upp föregående kommando räcker det att trycka på uppåtpiltangenten som sitter nere till höger på tangentbordet. Trycker du flera gånger bläddrar du successivt tillbaka bland gamla kommandon. Vill du veta mer om något kommando kan du ta fram ett manualblad med kommandot `man`, till exempel `man mkdir` för att få reda på mer om kommandot `mkdir`. Tryck ner tangenten `q` på tangentbordet om du vill avsluta `man`.

Användbara kommandon

Funktion	Kommandoexempel
Lista innehåll i aktuell katalog	<code>ls</code>
Lista innehåll i aktuell katalog, visa även gömda filer	<code>ls -a</code>
Byt aktuell katalog till hemkatalogen <code>~</code>	<code>cd</code>
Byt aktuell katalog till underkatalogen <code>lab1</code>	<code>cd lab1</code>
Byt aktuell katalog till "katalogen ovanför"	<code>cd ..</code>
Skapa en katalog som heter <code>lab1</code>	<code>mkdir lab1</code>
Kopiera filen <code>exempel</code> till din <code>lab1</code> -katalog	<code>cp /info/prgt/exempelfiler/labbar/exempel ~/prgo17/lab1/</code>
Kopiera filen <code>.emacs</code> till hemkatalogen	<code>cp /info/prgt/exempelfiler/.emacs ~/</code>
Ta bort filen <code>exempel</code> Obs! Filen försvinner för alltid	<code>rm exempel</code>
Ta bort katalogen <code>lab2</code> och alla underkataloger Obs! Innehållet försvinner för alltid	<code>rm -r lab2</code>
Exekvera programmet Pythonfilen	<code>python3 Pythonfilen</code>

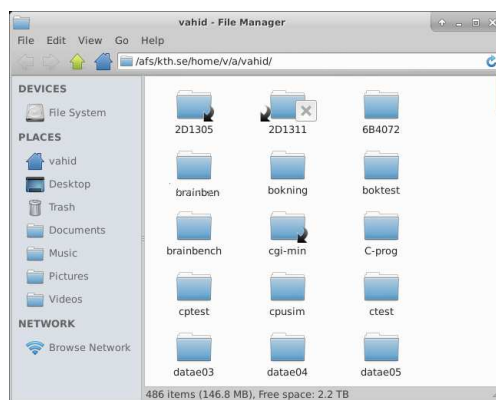
Filer och kataloger

Med en filhanterare kan du utforska datorns och nätverkets *kataloger*. En katalogs innehåll kan bestå av andra kataloger och *filer*. Kataloger som ligger i en annan katalog kallas ibland *underkataloger*.

Starta filhanteraren genom att klicka på andra ikonerna uppfifrån som visas i bilden på nästa sida.



Filhanteraren kommer att visa innehållet i din hemkatalog.



Redigeringsprogrammet Emacs

För att skapa och ändra filer används ett redigeringsprogram. Emacs är ett kraftfullt redigeringsprogram som har flertalet finesser om man t.ex. skall skriva *Pythonkällkod*. En av de viktigaste är *indentering*. Med indentering menas att texten i Pythonkällkoden skjuts in en bit beroende på vilken del av programmet som texten utgör.

Du startar Emacs genom att skriva **emacs&** i ett terminalfönster.

Innan du fortsätter med Emacs ska du skapa en ny katalog i **prgo17** och kalla den **lab1**. Kontrollera att katalogen **lab1** finns. Skapa sedan ytterligare en katalog i **prgo17** för varje laboration (**lab2**, ..., **lab5**) så blir det enklare för dig att hålla reda på filerna under kursens gång. Ladda ner sedan filen **exempel** från kursens hemsida till din katalog **lab1**. Ovanstående filer hittar man genom länken "Laboration 1" under "Modules>Vecka 1" på kursens hemsidan.

Filer och buffertar

Två viktiga begrepp i Emacs är *fil* och *buffert* ("file", "buffer"). En fil är något som finns sparad på en hårddisk. Om datorn slås av kommer en fil finnas kvar och kan utnyttjas när datorn startas igen. En buffert är något som används tillfälligt under tiden du skriver. Ändringarna i en buffert finns inte automatiskt kvar om datorn slås av. **Det är alltså viktigt** att spara ändringar i en buffert

till en fil lite då och då. Detta gäller speciellt om filen ska utnyttjas till något annat (t ex exekvering).

Öppna en existerande eller en ny fil

Öppna filen `exempel` som du kopierat tidigare. Man kan öppna en fil via menyn (**Files** → **Open File**) eller genom tangentkombinationer (**C-x C-f**, d v s håll ned kontrolltangente, tryck på **x**, håll ned kontrolltangente, tryck på **f**). Längst ned i Emacs dyker då en rad med den aktuella sökvägen upp. Ersätt den aktuella sökvägen med `~/prgo17/lab1/exempel`. Notera hur Emacs använder tecknet `/` för att separera katalognamn och filnamn, precis som i terminalfönstret. **Tips!** "TAB completion" kan även användas i emacs. När hela sökvägen är inskriven trycker du på returtangenten för att bekräfta valet. Innehållet i filen `exempel` ska nu dyka upp i en av Emacs buffertar och börja enligt:

Prinsessan av Babylonien

Det var en mörk vinterafton i den lilla stugan i Skrolycka. Kattrinna, hustrun i gården, satt och spann, och katten låg i hennes knä och spann, han också, så gott han kunde. Mannen, Jan Andersson, satt vid spisen och värmdes sig med ryggen mot elden. Han hade hela dagen gått och huggit ved i Erik i Fallas skog, så att ingen kunde begära, att han skulle ta sig före något arbete nu, när han var hemma. Inte en gång Kattrinna hade något att anmärka på att han nu inte gjorde annat än lekte och pratade med deras lilla flicka, som den här vintern gick på sitt femte år.

...

Observera att om den valda filen inte existerar, så kommer Emacs öppna en ny fil med det valda namnet. Med andra ord används med fördel **Files** → **Open File** eller **C-x C-f** både för att öppna en befintlig fil och för att skapa en ny!

Redigering och användbara kommandon

Så fort du redigerar lite i en buffert i Emacs så kommer buffertens innehåll inte stämma överens med den sparade filens innehåll. Detta visas i Emacs genom att markeringen ****** visas till vänster om filnamnet längst ned. Sparas bufferten i en fil så försvinner markeringen.

Börja med att spara den aktuella bufferten i en fil med namnet `exempel.txt`. Att spara till en fil med annat namn görs med **Files** → **Save Buffer As** eller **C-x C-w**. **Var noggrann** med STORA och små bokstäver!

Ändra någonting i bufferten. Notera hur markeringen ****** dyker upp. Spara bufferten (**Files** → **Save Buffer** eller **C-x C-s**) och notera hur markeringen ****** försvinner.

Du har nu använt några av de vanligaste kommandona i Emacs. Det finns otroligt många fler, men för denna kurs kan nedanstående kommandon vara bra att komma ihåg. Med **C-** avses kontrolltangente nedtryckt, med **M-** avses "meta"-tangente nedtryckt. På en UNIX-dator fungerar både "diamant"-tangente (höger om Kontrolltangente) och **Esc**-tangente (uppe till vänster) som metatangente. Trycker man **M-x** kan man skriva in namnet på ett emacs-kommando och köra det. Den som vill lära sig mer om emacs kan läsa Emacs tutorial som nås genom att trycka **F1 F1 t**.

Funktion	Menyval	Kommando
Avbryt på börjat kommando		C-g
Öppna befintlig fil/ skapa ny fil	Files → Open File...	C-x C-f
Spara buffert i fil	Files → Save Buffer	C-x C-s
Spara i fil under annat namn	Files → Save Buffer As...	C-x C-w
Stäng buffert	Files → Kill Current Buffer	C-x k
Ångra	Edit → Undo	C-_
Klipp ut från markören till slutet av raden (kan upprepas)		C-k
Start av markering		C-mellanslag
Slut av markering (kopiera)	Edit → Copy	M-w
Slut av markering (klipp ut)	Edit → Cut	C-w
Klistra in	Edit → Paste	C-y
Skriv ut aktuell buffert	Tools → Print → Print Buffer	
Sök	Search → Search...	C-s
Sök och byt	Search → Query Replace...	M-%

Python uppgift

Gör följande:

1. Öppna emacs
2. Skriv följande kod in i emacs:

```
print("Hej")
namn = input("Vad heter du?")
print("Ohhh", namn, "vilket fint namn!")
```
3. Spara filen som `fintnamn.py` i katalogen `prgo17/lab1/`
4. Öppna ett terminalfönster och se till att aktuell katalog är `prgo17/lab1`
5. Kör programmet med kommandot `python3 fintnamn.py`
6. Skriv kommentarer för varje rad av programkoden som förklarar hur programmet fungerar

Redovisning

Använd `Queue` för kursen `prgo` för att redovisa din laboration. Efter redovisningen kan du kontrollera att dina resultat har blivit rapporterade på sidan <http://rapp.csc.kth.se>, resultat av rapportering kommer att synas först när handledaren har rapporterat.

Tänk på att senare laborationerna tar betydligt längre tid än den schema-lagda datorsalstiden, så förbered dig och påbörja laborationen långt innan det

schemalagda laborationstillfället! Du har tillgång till CSCs datorsalar dygnet runt, men salarna kan vara bokade vissa tider under dagen. Behöver du hjälp, se länken hjälp på kurswebben.

Logga ut

För att logga ut kan du klicka på ikonen längst upp till höger på menyn.



När inloggningssidan visas betyder att du har loggat ut.

Kontroll nästa laboration

- ☐ Jag har registrerat mig på kursen på <http://rapp.csc.kth.se>.
- ☐ Jag vet hur jag hittar information på kurshemsidan.
- ☐ Jag vet hur man skapar, öppnar och sparar filer samt kan använda Emacs.
- ☐ Jag kan förklara hur pythonprogrammet som finns i filen "fintnamn.py" fungerar i detaljer.
- ☐ Jag vet hur jag kan kontrollera vilka laborationer har rapporterats godkänt genom <http://rapp.csc.kth.se>.