

Instruktioner för att kunna programmera på skolans datorer

De här instruktionerna är viktiga att känna väl till inför tentamen samt inför laborationsredovisningar. Många av er kommer att använda bärbara datorer i kursen, men några behöver kunna genomföra hela kursen på skolans datorer. Dessa instruktioner stödjer detta.

Man kan programmera på skolans datorer på två sätt, genom att köra en virtuell maskin (Windows XP) eller så kan man köra på skolans datorer direkt. Skolans datorer kör Windows 7 och har alla verktyg installerade som behövs. Instruktioner för hur man arbetar virtuellt finns på KTH – Socials forum om kursen HF1010, googla på det så hittar ni det stället. De som gått kursen HF1010 har redan arbetat virtuellt och har redan en start i det.

Vi ska nu beskriva vad man behöver göra på skolans datorer för att kunna programmera i C och inte göra detta virtuellt.

På skolans datorer finns det installerat tre verktyg, MinGW, msys och Emacs. Om man går till C: på datorn ser vi katalogerna MinGW och emacs-22.3. Katalogen MinGW innehåller en underkatalog som heter msys. Det är på det här sättet som de tre verktygen är installerade. För att vi ska kunna börja använda dem måste vi tala om för systemet att vi vill använda dem. Det första vi gör är att öppna en kommandoruta. Det kan man göra genom att klicka på den färgglada Windowsflaggan längst ner till vänster och skriva "cmd" i rutan där det står "Search programs and files". Då startar en svart ruta, det ser ut ungefär så här:

A screenshot of a Windows command prompt window. The title bar at the top reads "C:\Windows\system32\cmd.exe". The main area of the window is black with white text. The text displayed is: "Microsoft Windows [Version 6.1.7601] Copyright (c) 2009 Microsoft Corporation. All rights reserved. C:\Users\SOTTEST2.STH>". The window has standard Windows window controls (minimize, maximize, close) in the top right corner and a scrollbar on the right side.

Vi kan ge kommandon till systemet via denna ruta och allt eftersom kursen fortskrider kommer vi att bli mycket förtrogna med denna ruta. Att ge textkommandon till en dator är det klassiska sättet att hantera en dator och det är bra att kunna det. Grafiska gränssnitt, med fönster och kryssrutor etc är bra men har faktiskt begränsningar ibland.

Programmering innebär som ni fått höra på föreläsningar och kanske upplevt själva att först skapa en textfil med programmet i. Denna fil är i textformat, en så kallad textfil, vilket innebär att jag, som människa kan läsa den. Vi kan öppna den med ett enkelt texthanteringsprogram som notepad eller emacs (som vi ska se senare) och läsa innehållet. Ett C-program kan ha följande utseende:

```
#include <stdio.h>
main()
{
    printf("Hello World!\n");
}
```

Det är denna text som ska in i textfilen. När det gäller C-program ska den filen sluta på .c, inte på .txt eller liknande.

Det är gott och väl att skapa ett program, men innan vi kan köra det programmet måste vi kompilera och länka det. I miljön på skolan utförs det av ett verktyg som heter gcc. Dock, innan vi kan använda det verktyget måste vi, som vi nämnde tidigare tala om för systemet var det verktyget finns. Information om var körbara program finns (som gcc) sparas i så kallade miljövariabler. Det är variabeln path som anger var körbara program finns. För att se det nuvarande innehållet i path, skriv bara path vid kommandoprompten. Det kan se ut så här:

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\SOTTEST2.STH>path
PATH=C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Program Files\Common Files\Autodesk Shared\;C:\Program Files (x86)\Autodesk\Backburner\;C:\Program Files (x86)\Common Files\Autodesk Shared\;C:\Program Files\MATLAB\R2011a\runtime\win64;C:\Program Files\MATLAB\R2011a\bin;C:\MinGW\bin;C:\MinGW\msys\1.0\bin;C:\Program Files (x86)\Novell\ZENworks\bin

C:\Users\SOTTEST2.STH>_
```

I det här fallet har vi redan information om var programmeringsverktygen finns i systemet, vi ser det på att det förekommer texten "C:\MinGW\bin" (som anger var gcc finns) samt även texten "C:\MinGW\msys\1.0\bin" (som anger var lite mer verktyg finns.) Om dessa texter inte finns i path så kan man inte utföra en kompilering med verktyget gcc. Ett enkelt test för att se om information finns i path är (förutom det ovanstående) att helt enkelt prova att skriva gcc vid kommandoprompten, vi ska då få responsen "gcc: no input files". Det betyder att vi anropat kompilatorn, hittat den, men kompilatorn själv klagar på att den inte fått något program att kompilera. Om inte path är satt skulle vi fått en respons i stil med "'gcc' is not recognised as an internal or external command, operable program or batch file." Då måste vi göra följande. Vi skriver vid kommandoprompten:

```
set path=%path%;C:\MinGW\bin;C:\MinGW\msys\1.0\bin
```

Det ska utläsas så här:

"Sätt path till det nuvarandeinnehållet i path och lägg till ;C:\MinGW\bin;C:\MinGW\msys\1.0\bin".

Då har vi en fungerande kommandoruta som har kontakt med kompilatorn. Nästa steg är att se till att vi inte spara saker lokalt på den dator vi sitter vid. Det uppnås genom att vi skriver H: vid prompten. Prompten byter då utseende till H:\> för att indikera nuvarande arbetskatalog. Med kommandot dir kan vi sedan få en lista på katalogerna i H. Det kan se ut så här:

```
C:\Users\SOTTEST2.STH>h:
```

```
H:\>dir
```

```
Volume in drive H is VOL1  
Volume Serial Number is 5C56-4F4C
```

```
Directory of H:\
```

```
2007-03-22 12:12 <DIR>      .  
2007-03-22 12:12 <DIR>      ..  
2011-01-20 17:24 <DIR>      .metadata  
2007-03-23 15:18          25 872 Drawing1.dwg  
2011-03-29 15:22          3 822 232 O980-A07.EXE  
2009-09-28 19:47 <DIR>      Profile  
2011-01-20 17:26 <DIR>      test2  
2011-07-21 15:56 <DIR>      testmapp  
2011-03-30 11:27 <DIR>      VicoClient  
2009-09-28 19:47 <DIR>      Windows NT 5.1 Workstation Profile  
                2 File(s)  3 848 104 bytes  
                8 Dir(s) 195 686 621 184 bytes free
```

```
H:\>
```

Observera att vi, efter varje fullbordat commando får tillbaka prompten som ett tecken på att system är berett att ta emot ett nytt commando. Här kan det vara bra att skapa en ny katalog. Man kan göra det grafiskt eller skriva, vid prompten "md cprog". Kommandot "md" betyder "make directory" och det är mycket rekommenderat att ni använder kataloger för att organisera allt ert arbete efter hand. Så skriver vi cd cprog för att gå ner i den nya katalogen. Vi kan också skriva dir där och då får vi upp en tom lista, vi har ju inte skapat några filer ännu. Det ser ut så här:

```
H:\>md cprog
```

```
H:\>cd cprog
```

```
H:\cprog>dir
```

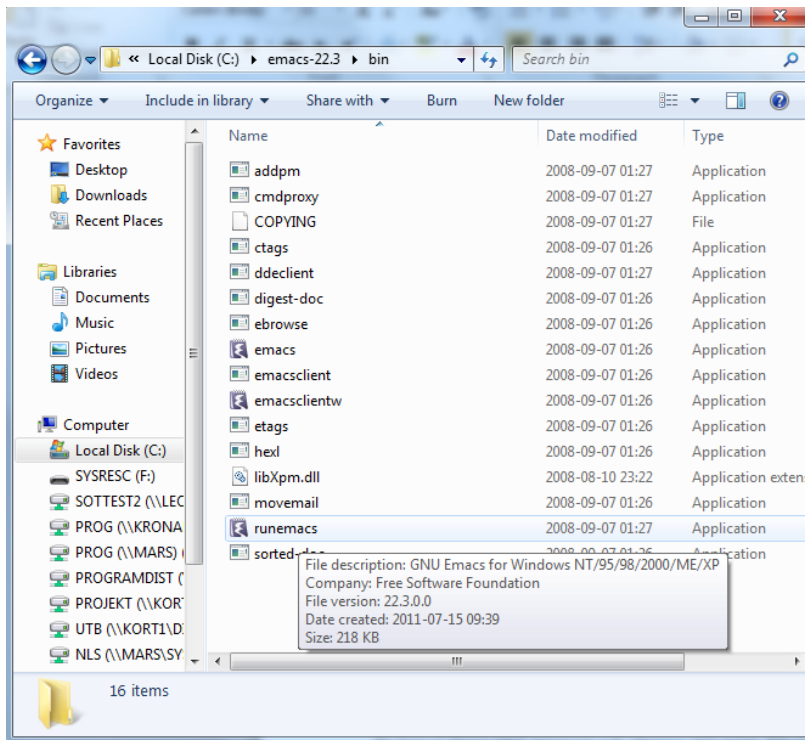
```
Volume in drive H is VOL1  
Volume Serial Number is 5C56-4F4C
```

```
Directory of H:\cprog
```

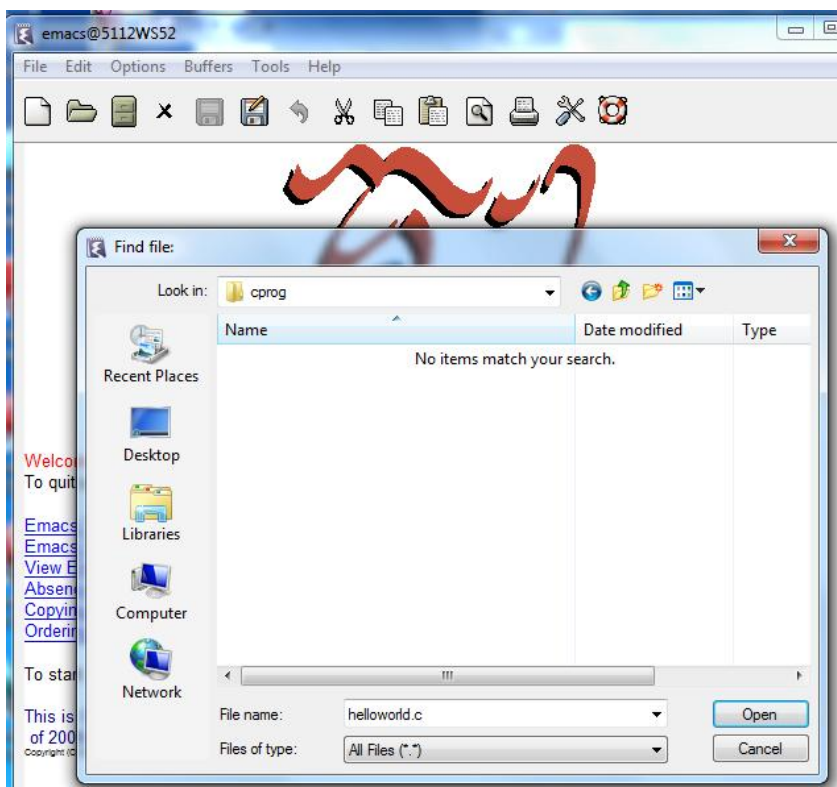
```
2011-08-29 17:06 <DIR>      .  
2011-08-29 17:06 <DIR>      ..  
                0 File(s)      0 bytes  
                2 Dir(s) 195 686 621 184 bytes free
```

```
H:\cprog>
```

Nu kan vi skapa ett C-program och placera i H:\cprog. Kör programmet emacs genom att öppna C: och gå in i katalogen emacs-22.3 och starta emacs. Det ser ut så här:

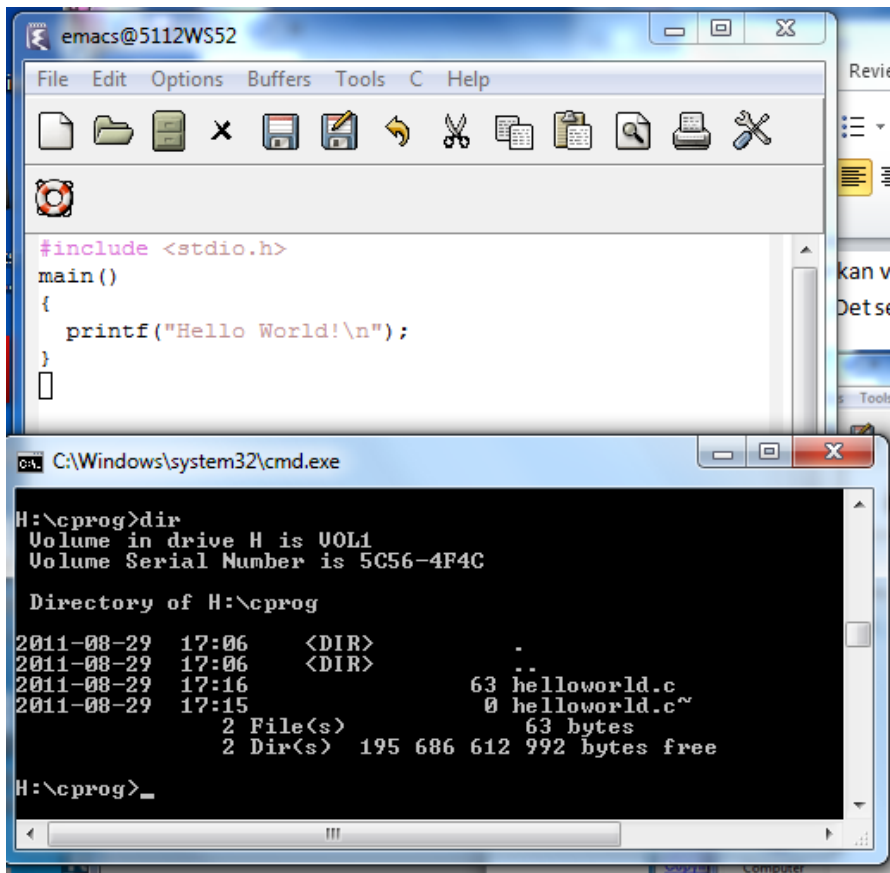


När emacs startar kan vi skapa en ny fil på H:\cprog genom att välja File->Visit New File och välja platsen H:\cprog. Det ser ut så här:

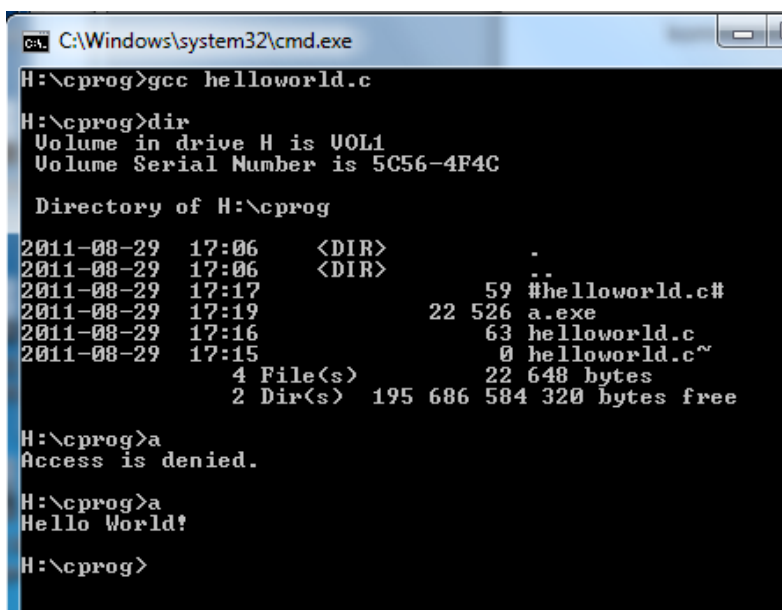


Lägg märke till att filnamnet, helloworld.c, slutar på.c. Mycket viktigt!

Vi redigerar filen så att den innehåller HelloWorld-programmet ovan och sparar. Om vi ger kommandot dir vid prompten har vi följande utseende:



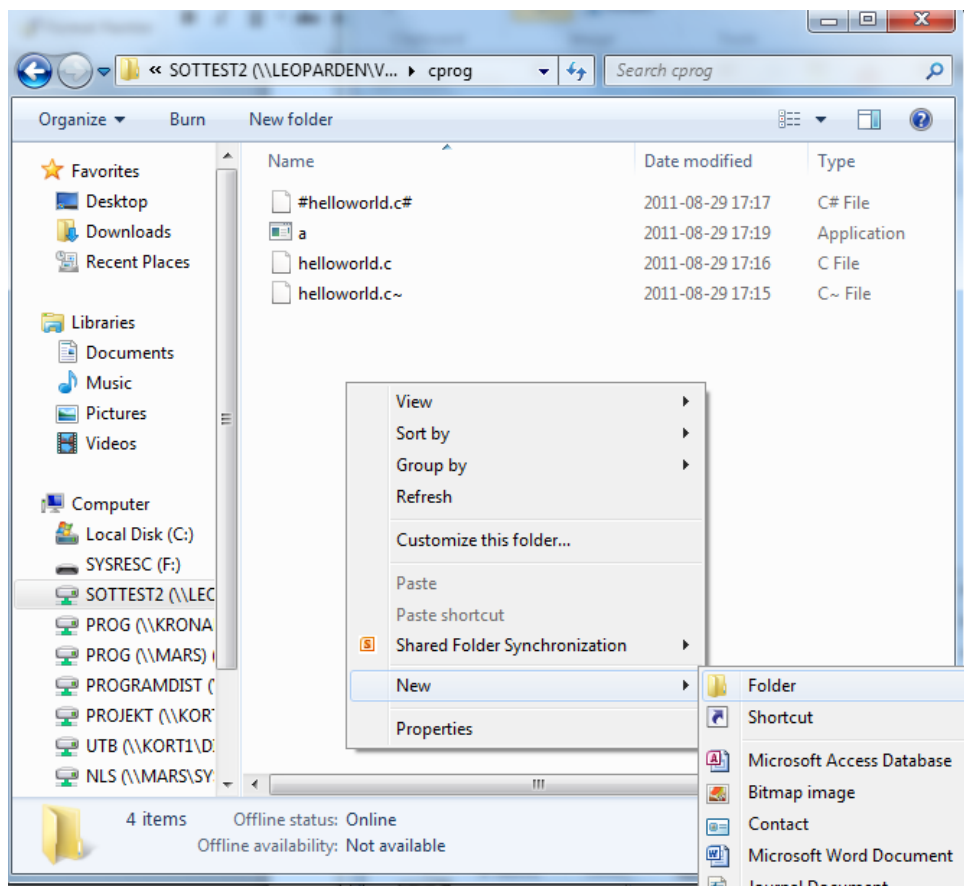
Vi har alltså C-programmet, skrivet med Emacs, och vi ska nu hantera det programmet med kommandoprompten. Vi ska kompilera programmet. Det gör vi genom att skriva gcc helloworld.c. Vi kör sedan en dir för att se vilka förändringar som uppkommit. Det har skapats en till fil, a.exe, som är det körbara programmet. Det ser ut så här:



För att köra programmet kan vi skriva, bara "a", eller "a.exe" som syns ovan. Windows tillåter dock inte körning av program hur som helst så vi får en grafisk dialog om att programmet måste tillåtas att köras. Andra gången vi kör programmet (efter att ha svarat "ja, det är OK att köra programmet") så ser vi dock att programmet körs, vi får se utskriften "Hello World!" som kommer ut i den svarta rutan.

Den nya filen, som heter a.exe, är en så kallad körbar fil, eller program. Det är en fil, precis som helloworld.c, men den är i *binärt format* och innehållet är maskinkod som processorn kan läsa direkt. En människa kan inte läsa sådan kod utan avancerade hjälpmedel, och vi ska aldrig läsa maskinkod i den här kursen.

Med dessa enkla instruktioner har du lärt dig grunderna i hur man programmerar på skolans datorer. Det kan också vara värt att understödja manövrarna med det grafiska gränssnittet, för att till exempel skapa en underkatalog till katalogen H:\cprog kan man gå in där grafiskt och högerklicka och sen välja New->Folder... Så här:



Här ser vi också grafiska presentationer av tidigare filerna vi arbetade med, "helloworld.c" och "a.exe".

Det kan nu vara värt att gå igenom en enkelt tutorial om Emacs och även läsa lite om kommandoprompten, googla till exempel på "command prompt Windows", här hittar man till exempel <http://commandwindows.com>.