

# DD1311 Programmeringsteknik för CL1

## Laborationer läsåret 2008-2009

Fyll i ditt namn och personnummer med bläck eller motsvarande.

Kursledare är Linda Kann, linda@nada.kth.se.

Namn ..... Personnr .....

### Laborationer

Labb 1	Godkänt den ..... (2009-01-23)	Kvitteras .....	Timmar .....
Labb 2	Godkänt den ..... (2009-01-30)	Kvitteras .....	Timmar .....
Labb 3	Godkänt den ..... (2009-02-06)	Kvitteras .....	Timmar .....
Labb 4	Godkänt den ..... (2009-02-13)	Kvitteras .....	Timmar .....
Labb 5	Godkänt den ..... (2009-02-20)	Kvitteras .....	Timmar .....

### Prov

Prov	Godkänt den ..... (2009-03-03)	Kvitteras .....
------	-----------------------------------	-----------------

### P-del

Spec	Godkänt den ..... (2009-03-16)	Kvitteras .....	
Prototyp	Godkänt den ..... (2009-03-19)	Kvitteras .....	
Granskning	Godkänt den ..... (2009-04-28)	Kvitteras .....	
Redovisning	Godkänt den ..... (2009-04-28)	Kvitteras .....	Timmar ..... A B C D E F

## Laboration 1

Kurshemsidan, hederskodex, incheckning, IDLE, print-satsen, kommentarer, variabler, strängar, beräkningar, indata

*Denna labb redovisas på veckans gruppmöte. Ta med utskrifter av dina program samt dina svar på instuderingsfrågorna. Vid redovisningen ska du kunna förklara både program och svar för dina kurskamrater och din assistent.*

### Hitta dator och labbkompis

Vilken dator du ska sitta vid står på den lila lappen du fick av kursledaren. De inledande fem labbarna gör ni i grupper om två och din labbkompis är den som har fått samma datornummer på sin lila lapp.

### Inloggning

Sätt dig vid en dator tillsammans med din labbkompis och välj Java Desktop System under knappen Options/Sessions och logga sedan in. Skärmen kommer att se ut ungefär så här:



Längst ned finns frontpanelen som motsvarar ”Taskbar” i Windows. Den används för att starta vissa program, ändra inställningar m.m. Läs gärna mer om den i Unixhäftet (säljs på CSCs studentexpedition). Klicka med högra musknappen på skrivbordet. Då får du en popup-meny där du kan bl.a. kan starta ett terminalfönster. Gör det! När du skriver i det fönstret ger du kommandon till kommandotolken. Kom ihåg att kommandotolken skiljer mellan stora och små bokstäver.

### Webbläsare och kursinformation

*Mozilla* är en webbläsare som startas genom att välja **Launch**→**Applications**→**Internet**→**Web Browser**. Klicka en gång i det fönster du nu får upp och skriv in följande URL-adress i adressfältet:

<http://www.csc.kth.se/DD1311/>

Klicka på länken "prgcl09". Du ska nu få upp en sida med kursinformation för programmeringsteknikkursen för CL1. Denna sida är viktig! Här kommer vi att lägga ut information under kursens gång, så vänj dig vid att alltid titta på sidan när du loggar in.

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**.

### Hederskodex

På Nada används en gemensam hederskodex för alla Nadakurser. Du hittar den via kurshemsidan och är själv ansvarig att läsa igenom och följa den.

Vilka är de fem reglerna i hederskodexen?

### Checka in på kursen

På kurshemsidan finns information om hur man checkar in på kursen. Se till att båda blir incheckade. Var noggrann med alla personuppgifter! På den lila lappen står det vilken grupp ni tillhör.

### Användbara kommandon i Unix


Terminalfönstret har en mängd kommandon och nedan finns exempel på de mest nödvändiga för denna kurs.

Funktion	Kommandoexempel
Skapa en katalog som heter <code>prgcl09</code>	<code>mkdir prgcl09</code>
Lista innehåll i aktuell katalog	<code>ls</code>
Byt aktuell katalog till underkatalogen <code>prgcl09</code>	<code>cd prgcl09</code>
Byt aktuell katalog till hemkatalogen <code>~</code>	<code>cd</code>
Byt aktuell katalog till "katalogen ovanför"	<code>cd ..</code>
Ta bort filen <code>hej.py</code> <b>Obs!</b> Filen försvinner för alltid	<code>rm hej.py</code>
Köra Pythonprogram	<code>python hej.py</code>
Skriv ut filen <code>hej.py</code> till skrivaren	<code>print hej.py</code>

## Terminalfönstret

Alla kommandon ovan skriver man i terminalfönstret. Till exempel kan du lista alla filer i din katalog med kommandot `ls` (list).

Det du framför allt behöver veta är hur du flyttar dig mellan olika kataloger i terminalfönstret. För att gå ner i katalogen `prgc109` skriver du `cd prgc109`, vilket utläses *change directory to prgc109*. För att gå upp en nivå skriver du `cd ..` (glöm inte mellanslaget före punkterna).

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

## Starta IDLE

Välj `Start` → `Programs` → `Python` → `IDLE` från startmenyn. Fönstret `Python Shell` som öppnas är en Python-interpretator, där man direkt kan skriva in en sats och se vad som händer.

Längst upp i fönstret kan man välja bland menyerna `File`, `Edit`, `Windows` och `Help`. Välj `File` → `New Window` så öppnas ett nytt fönster där man kan skriva in, redigera, spara, och köra Python-program.

För att spara det program du skrivit in väljer du `File` → `Save As`. Ge alltid dina program suffixet `.py` (t ex `Gorn.py`).

När du är redo att provköra programmet väljer du `Edit` → `Run Script`. Utmatningen från programmet kommer att synas i `Python Shell`-fönstret.

## Skrivaren

Skrivarna på CSC heter oftast samma som salen de är placerade i. Kommandot som skriver ut är `print`. Om du vill skriva ut filen `hej.py` skriver du `print hej.py` i terminalfönstret *Spara papper! Skriv inte ut i onödan!*

## Program

Krav för programmen i labb 1: Visa att du kan hantera variabler, tilldelning, inläsning, utskrift, enkla beräkningar.

1. Skriv ett program som beräknar dricks. Användaren får mata in restaurangnotans summa och programmet ska skriva ut 15 respektive 20 procent av summan. (Dawson s.50)
2. Skriv ett program som konverterar temperaturer i Celsius till Fahrenheit med formeln  $F = \frac{9}{5}C + 32$ . Användaren får mata in en temperatur i Celsius och programmet ska skriva ut temperaturen omvandlad till Fahrenheit. (Gaddis s. 74)

## Instuderingsfrågor

1. Vilka programmeringsspråk känner du till?

2. Hur skriver man en kommentar i Python?
3. Vad är IDLE?
4. Ge exempel på två olika datatyper.
5. Hur konverterar man mellan olika datatyper?
6. Vad är en variabel?
7. Hur ger man en variabel ett värde?
8. Hur läser man in ett ord till programmet?
9. Hur läser man in ett tal till programmet?
10. Vad händer när en metod anropas?
11. Hur får man programmet att skriva ut text på skärmen?
12. Hur får man programmet att skriva ut ett variabelvärde på skärmen?
13. Vad är det för vits med att låta programmet läsa in värden?
14. Vad menar man när man säger att ett program är användarvänligt?

### **Efter labben**

Hur tycker du att ditt eget arbete med labben fungerade? Borde du ha gjort något annorlunda? Fundera en stund, och fyll sedan i rutan på nästa sida under rubriken "Tips till mig själv inför labben".

### **Logga ut**

För att logga ut väljer man `logout` från Launch-knappen. När inloggningssidan visas betyder att du har loggat ut.

## Laboration 2

Slumptal, if-satsen, while-slingan, programplanering, for-slingan, strängar, tupler

*Denna labb redovisas på veckans gruppmöte. Ta med utskrifter av dina program samt dina svar på instuderingsfrågorna. Vid redovisningen ska du kunna förklara både program och svar för dina kurskamrater och din assistent.*

### Tips till mig själv inför labben:

### Program

Krav för programmen i labb 2: Visa att du kan hantera if/else-satser, for-slingan och while-slingan.

1. Skriv ett program som läser in en persons längd (i meter) och vikt (i kg) och beräknar  $BMI = \frac{vikt}{längd * längd}$ . Sedan ska programmet skriva ut om personen är underviktig (BMI under 18.5), överviktig (BMI över 25) eller normalviktig. (Gaddis s. 150)
2. Skriv ett program som slumpar hundra myntkast och talar om hur många kast som gav krona och hur många som gav klave. (Dawson s. 87)
3. Skriv ett program som slumpar fram en matteuppgift (t ex *Vad blir 8\*7?*) och ställer om frågan tills användaren ger rätt svar.

### Instuderingsfrågor

1. Hur slumpar man ett tal i ett Pythonprogram?
2. Vad gör raden `import random`
3. Vad är det för skillnad på if-satsen och while-slingan?
4. Vad är en oändlig slinga?
5. Vad innebär det att två slingor är nästlade?
6. Ge ett exempel på ett villkor. Vilka värden kan ett villkor anta?
7. Hur fungerar operatorn `and` i ett villkor?
8. Vad är en algoritm?
9. När använder man en for-slinga istället för en while-slinga?

10. Hur skriver man ut första bokstaven i en sträng?
11. Hur ändrar man första bokstaven i en sträng?
12. Vad är en tuppel?
13. Hur slumpar man ett ord ur en tuppel av ord?
14. Vad är ett index i en sträng eller tuppel?

## Laboration 3

Listor, uppslagslistor, funktioner, parametrar, namnrymden.

*Denna labb redovisas på veckans gruppmöte. Ta med utskrifter av dina program samt dina svar på instuderingsfrågorna. Vid redovisningen ska du kunna förklara både program och svar för dina kurskamrater och din assistent.*

### Tips till mig själv inför labben:

### Program

Krav för programmen i labb 3:

Programmen ska delas upp i funktioner med parametrar och returvärden.

1. Skriv först två funktioner:

- *medel* som tar tre provresultat som indata (parameter) och beräknar medelvärdet (returvärde)
- *betyg* som tar ett provresultat som indata och returnerar ett bokstavs-betyg i intervallet A-F (välj själv betygsskala).

Skriv sedan ett program som ber användaren att mata in tre provresultat och använd funktionerna för att skriva ut betyg på varje prov och på hela kursen. (Gaddis s. 222)

2. Ändra i programmet du nyss skrev så att det fungerar för ett godtyckligt antal provresultat. Tips: läs in provresultaten i en lista, och låt den vara parameter till *medel*

### Instuderingsfrågor

1. Vad är det för skillnad mellan en lista och en tuppel?
2. Beskriv tre metoder som finns för en lista?
3. Vad är en nästlad lista?
4. Visa hur man skriver ut alla elementen i en nästlad lista.
5. Rita upp ett exempel som visar hur referenser fungerar.
6. Vad kan man använda en ordlista (dictionary) till?
7. Beskriv tre metoder som finns för en ordlista.



8. Hur definierar man en funktion?
9. Varför anropar man funktioner?
10. Vad innebär abstraktion?
11. Hur tar en funktion emot indata? Visa med ett exempel!
12. Hur skickar man utdata från en funktion? Visa med ett exempel!
13. Vad blir returvärdet om man inte skrivit att något ska returneras?
14. Vad är det för skillnad mellan lokala och globala variabler?

## Laboration 4

Filer, särfall(exceptions), konservering (pickling), klasser, metoder, objekt, konstruktor, attribut

*Denna labb redovisas på veckans gruppmöte. Ta med utskrifter av dina program samt dina svar på instuderingsfrågorna. Vid redovisningen ska du kunna förklara både program och svar för dina kurskamrater och din assistent.*

### Tips till mig själv inför labben:

### Program

Krav för programmen i labb 4:

Programmen ska delas upp i funktioner med lämpliga parametrar och returvärdet. Använd slingor när något ska upprepas.

1. Skriv ett program som ber användaren om ett filnamn. Programmet ska sedan visa upp varje rad ur filen på skärmen, med radnumret först på raden. Exempel:  
1: Det var en mörk och stormig natt,  
2: och från slottet på kullen hördes mystiska ljud...  
(Gaddis s. 272)
2. Skriv ett program som simulerar en tv-apparat genom att skapa ett tv-objekt. Användaren ska kunna byta kanal samt höja och sänka volymen. Se till att kanalnummer och volym håller sig inom givna gränser. (Dawson s. 251)

### Instuderingsfrågor

1. I vilka moder kan man öppna en textfil?
2. Ge två olika förslag på hur man kan läsa in alla rader från en fil.
3. Varför vill man låta ett program skriva ut data på en fil?
4. Vilka typer av data kan man konservera (pickle)?
5. Hur återställer man konserverade data?
6. När används särfall (exception)?
7. Ge exempel på ett inbyggt särfall!
8. Vad är relationen mellan klass och objekt?

9. Vad är en metod?
10. Hur fungerar en konstruktor?
11. Hur kommer man åt ett objekts attribut?
12. Vad kännetecknar klassattribut och statiska metoder?
13. Varför ska man använda metoder för att ändra ett objekts attribut?
14. Vad händer om man har två understreck i början av metodnamnet?

## Laboration 5

Objektorienterad programmering, arv, polymorfism, moduler

*Denna labb redovisas på veckans gruppmöte. Ta med utskrifter av dina program samt dina svar på instuderingsfrågorna. Vid redovisningen ska du kunna förklara både program och svar för dina kurskamrater och din assistent.*

**Tips till mig själv inför labben:**

### Program

Krav för programmen i labb 5:

Klassen ska ha lämpliga attribut och metoder.

Använd *olika* namn för attributen och metoderna.

Kommentera alla metoder.

1. Hitta på en egen klass
  - Klassen ska ha flera attribut och metoder.
  - Skapa ett objekt och testa att metoderna fungerar.
2. Skapa flera objekt
  - Skriv först ihop en textfil med data för flera objekt.
  - Låt programmet läsa in data från textfilen...
  - ... och skapa objekt som läggs i en lista.
  - Låt till sist programmet skriva ut alla objekten.

### Instuderingsfrågor

1. Föreslå något som kan representeras med en klass.
2. Vilka attribut ska din klass ha?
3. Och vilka metoder?
4. Rita ett objekt (med exempeldata) som är en instans av din klass.
5. Vad används metoderna `__str__` och `__cmp__` till?
6. Hur går det till när ett objekt skickar ett meddelande till ett annat?
7. Föreslå ett program där objekt behöver interagera.

8. Hur strukturerar man många objekt av samma typ i sitt program?
9. Rita ett exempel på en sådan struktur.
10. Hur lägger man upp en utskrift av data från alla objekten?
11. Vad används arv (inheritance) till?
12. Hur skriver man om man vill att en klass ska ärva från en annan?
13. Ge ett exempel på polymorfism.
14. Hur gör man en egen modul?