

# Programmering, grundkurs

## Kompendium

EDAA45, Lp1-2, HT 2016  
Datavetenskap, LTH  
Lunds Universitet

<http://cs.lth.se/pgk>

*Editor:* Björn Regnell

*Contributors:* ...

*Home:* <https://cs.lth.se/pgk>

*Repo:* <https://github.com/lunduniversity/introprog>

This manuscript is on-going work. Contributions are welcome!

*Contact:* [bjorn.regnell@cs.lth.se](mailto:bjorn.regnell@cs.lth.se)

*LICENCE:* CC BY-SA 4.0

<http://creativecommons.org/licenses/by-sa/4.0/>

Copyright © Computer Science, LTH, Lund University. 2016. Lund. Sweden.

# Framstegsprotokoll

## Genomförda övningar

Till varje laboration hör en övning med uppgifter som utgör förberedelse inför labben. Du behöver minst behärska de grundläggande övningarna för att klara labben inom rimlig tid. Om du känner att du behöver öva mer på grunderna, gör då även extrauppgifterna. Om du vill fördjupa dig, gör fördjupningsuppgifterna som är på mer avancerad nivå. Genom att du kryssar för nedan vilka övningar du har gjort, blir det lättare för handledaren att förstå vilka förkunskaper du har inför labben.

Övning	Grundläggande	Extra	Fördjupning
expressions	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
statements	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
functions	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
data	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
vectors	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
classes	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
traits	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
matching	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
matrices	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
sorting	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
scalajava	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
threads	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

## Godkända obligatoriska moment

För att bli godkänd på laborationsuppgifterna och inlämningsuppgiften måste du lösa deluppgifterna och diskutera dina lösningar med en handledare. Denna diskussion är din möjlighet att få feedback på dina lösningar. Ta vara på den! Se till att handledaren noterar när du blivit godkänd på detta blad, som är ditt kvitto. Spara detta blad tills du fått slutbetyg i kursen.

Namn: .....

Namnteckning: .....

Uppgift	Datum gk	Handledares namnteckning
kojo	.....	.....
simplewindow	.....	.....
files	.....	.....
cardgame	.....	.....
shapes	.....	.....
turtlerace-team	.....	.....
newlab-team	.....	.....
maze	.....	.....
bank-team	.....	.....
scalajava-team	.....	.....
life	.....	.....
Inl.Uppg.	.....	.....

### Inlämningsuppgift (välj en)

- ( ) bank
- ( ) mandelbrot
- ( ) draw
- ( ) egendefinerad

*Om egen, ge kort beskrivning:*

# Förord

Programmering är inte bara ett sätt att ta makten över systemen som människan skapar för att styra vårt samhälle. Det är också ett kraftfullt verktyg för tanken. Med kunskap i programmeringens grunder kan de som vill påbörja en livslång läranderesor som systemutvecklare och abstraktionskonstnärer. Programmeringsspråk och utvecklingsverktyg kommer och går, men de grundläggande koncepten bakom *all* mjukvara består: sekvens, alternativ, repetition och abstraktion.

Detta kompendium utgör kursmaterial för en grundkurs i programmering, med syfte att ge en solid bas för ingenjörstudenter och andra som vill utveckla system med mjukvara. Materialet omfattar en termins studier på kvartsfart och förutsätter kunskaper motsvarande gymnasienivå i svenska, matematik och engelska.

Kompendiet är framtaget för och av studenter och lärare, och distribueras som öppen källkod. Det får användas fritt så länge erkännande ges och eventuella ändringar publiceras under samma licens som ursprungsmaterialet. På kursens hemsida [cs.lth.se/pgk](http://cs.lth.se/pgk) och repo [github.com/lunduniversity/introprog](https://github.com/lunduniversity/introprog) finns instruktioner om hur du kan bidra till kursmaterialet.

Läromaterialet fokuserar på lärande genom praktiskt programmeringsarbete och innehåller övningar och laborationer som är organiserade i moduler. Varje modul har ett tema och en teoridel i form av föreläsningsbilder med tillhörande anteckningar.

I kursen används språken Scala och Java för att illustrera grunderna i imperativ och objektorienterad programmering, tillsammans med elementär funktionsprogrammering. Mer avancerad objektorientering och funktionsprogrammering lämnas till efterföljande fördjupningskurser.

Den kanske viktigaste framgångsfaktorn vid studier i programmering är att bejaka din egen upptäckarglädje och experimentlusta. Det fantastiska med programmering är att dina egna intellektuella konstruktioner faktiskt *gör* något som just *du* har bestämt! Ta vara på det och prova dig fram genom att koda egna idéer – det är kul när det funkar men minst lika lärorikt är felsökning, bugggrättande och alla misslyckade försök som efter hårt arbete vänds till lyckade lösningar och/eller bestående lärdomar.

Välkommen i programmeringens fascinerande värld och hjärtligt lycka till med dina studier!

*LTH, Lund 2016*



# Innehåll

<b>Framstegsprotokoll</b>	<b>3</b>
<b>Förord</b>	<b>5</b>
<b>I Om kursen</b>	<b>5</b>
<b>Kursens arkitektur</b>	<b>7</b>
0.0.1 hej . . . . .	8
<b>Anvisningar</b>	<b>9</b>
Föreläsningar . . . . .	9
Övningar . . . . .	9
Laborationer . . . . .	9
Resurstider . . . . .	9
Kontrollskrivning . . . . .	9
Tentamen . . . . .	9
<b>Hur lära att programmera?</b>	<b>11</b>
Lära genom att göra . . . . .	11
Vilken är din lärandestil? . . . . .	11
<b>Hur bidra till kursmaterialet?</b>	<b>13</b>
<b>II Moduler</b>	<b>15</b>
<b>1 Introduktion</b>	<b>17</b>
1.1 Vad är programmering? . . . . .	17
1.2 Hur fungerar en dator? . . . . .	18
1.3 Uttryck, värde och typ . . . . .	18
1.3.1 Uttryck med enkla typer . . . . .	18
1.3.2 Alternativuttryck . . . . .	18
1.4 Variabler, val, var . . . . .	18
1.5 Övning: expressions . . . . .	19
1.5.1 Grundläggande uppgifter . . . . .	19
1.5.2 Extrauppgifter: öva mer på grunderna . . . . .	21
1.5.3 Fördjupningsuppgifter: avancerad nivå . . . . .	21

1.6	Laboration: Quiz . . . . .	23
1.6.1	Obligatoriska uppgifter . . . . .	23
1.6.2	Frivilliga extrauppgifter . . . . .	23
<b>2</b>	<b>Kodstrukturer</b>	<b>25</b>
2.1	Vad är programmering? . . . . .	25
2.2	Hur fungerar en dator? . . . . .	26
2.3	Övning: Hello . . . . .	27
2.3.1	Grundläggande uppgifter . . . . .	27
2.3.2	Extrauppgifter: öva mer på grunderna . . . . .	28
2.3.3	Fördjupningsuppgifter: avancerad nivå . . . . .	28
2.4	Laboration: Quiz . . . . .	29
2.4.1	Obligatoriska uppgifter . . . . .	29
2.4.2	Frivilliga extrauppgifter . . . . .	29
<b>3</b>	<b>Funktioner, Objekt</b>	<b>31</b>
3.1	Vad är programmering? . . . . .	31
3.2	Hur fungerar en dator? . . . . .	32
3.3	Övning: Hello . . . . .	33
3.3.1	Grundläggande uppgifter . . . . .	33
3.3.2	Extrauppgifter: öva mer på grunderna . . . . .	34
3.3.3	Fördjupningsuppgifter: avancerad nivå . . . . .	34
3.4	Laboration: Quiz . . . . .	35
3.4.1	Obligatoriska uppgifter . . . . .	35
3.4.2	Frivilliga extrauppgifter . . . . .	35
<b>4</b>	<b>Datastrukturer</b>	<b>37</b>
4.1	Vad är programmering? . . . . .	37
4.2	Hur fungerar en dator? . . . . .	38
4.3	Övning: Hello . . . . .	39
4.3.1	Grundläggande uppgifter . . . . .	39
4.3.2	Extrauppgifter: öva mer på grunderna . . . . .	40
4.3.3	Fördjupningsuppgifter: avancerad nivå . . . . .	40
4.4	Laboration: Quiz . . . . .	41
4.4.1	Obligatoriska uppgifter . . . . .	41
4.4.2	Frivilliga extrauppgifter . . . . .	41
<b>III</b>	<b>Appendix</b>	<b>43</b>
<b>A</b>	<b>Terminalfönster och kommandoskal</b>	<b>45</b>
A.1	Vad är ett terminalfönster? . . . . .	45
A.2	Några viktiga terminalkommando . . . . .	45
<b>B</b>	<b>Editera</b>	<b>47</b>
B.1	Vad är en editor? . . . . .	47
B.2	Välj editor . . . . .	47



<b>C</b>	<b>Kompilera och exekvera</b>	<b>49</b>
C.1	Vad är en kompilator? . . . . .	49
C.2	Java JDK . . . . .	49
C.2.1	Installera Java JDK . . . . .	49
C.3	Scala . . . . .	49
C.3.1	Installera Scala-kompilatorn . . . . .	49
C.4	Read-Evaluate-Print-Loop (REPL) . . . . .	49
C.4.1	Scala REPL . . . . .	49
<b>D</b>	<b>Dokumentation</b>	<b>51</b>
D.1	Vad gör ett dokumentationsverktyg? . . . . .	51
D.2	scaladoc . . . . .	51
D.3	javadoc . . . . .	51
<b>E</b>	<b>Integrerad utvecklingsmiljö</b>	<b>53</b>
E.1	Vad är en IDE? . . . . .	53
E.2	ScalaIDE och Eclipse . . . . .	53
E.2.1	Installera ScalaIDE . . . . .	53
E.3	Handledning ScalaIDE/Eclipse . . . . .	53
<b>F</b>	<b>Byggverktyg</b>	<b>55</b>
F.1	Vad gör ett byggverktyg? . . . . .	55
F.2	Byggverktyget sbt . . . . .	55
<b>G</b>	<b>Versionshantering</b>	<b>57</b>
G.1	Vad är versionshantering? . . . . .	57
G.2	Versionshanteringsverktyget git . . . . .	57
<b>H</b>	<b>Lösningförslag till övningar</b>	<b>59</b>
H.1	Övning: hello . . . . .	60
H.1.1	Grundläggande uppgifter . . . . .	60
H.1.2	Extrauppgifter: öva mer på grunderna . . . . .	60
H.1.3	Fördjupningsuppgifter: avancerad nivå . . . . .	60
H.2	Övning: hello . . . . .	61
H.2.1	Grundläggande uppgifter . . . . .	61
H.2.2	Extrauppgifter: öva mer på grunderna . . . . .	61
H.2.3	Fördjupningsuppgifter: avancerad nivå . . . . .	61
H.3	Övning: hello . . . . .	62
H.3.1	Grundläggande uppgifter . . . . .	62
H.3.2	Extrauppgifter: öva mer på grunderna . . . . .	62
H.3.3	Fördjupningsuppgifter: avancerad nivå . . . . .	62
<b>I</b>	<b>Ordlista</b>	<b>63</b>



# **Del I**

## **Om kursen**



# Kursens arkitektur

<i>W</i>	<i>Modul</i>	<i>Övn</i>	<i>Lab</i>
W01	Introduktion	expressions	kojo
W02	Kodstrukturer	statements	–
W03	Funktioner, Objekt	functions	simplewindow
W04	Datastrukturer	data	files
W05	Vektoralgoritmer	vectors	cardgame
W06	Klasser, Likhet	classes	shapes
W07	Arv, Gränssnitt	traits	turtlerace-team
KS	KONTROLLSKRIVN.	–	–
W08	Mönster, Undantag	matching	newlab-team
W09	Matriser	matrices	maze
W10	Sökning, Sortering	sorting	bank-team
W11	Scala och Java	scalajava	scalajava-team
W12	Trådar, Web, Android	threads	life
W13	Design	Uppsamling	Inl.Uppg.
W14	Tentaträning	Extenta	–
T	TENTAMEN	–	–

Kursen består av ett antal moduler med tillhörande teori, övningar och laborationer. Genom att göra övningarna bearbetar du teorin och förbereder dig inför laborationerna. När du klarar laborationen i varje modul är du redo att gå vidare till efterkommande modul.

## Vad och hur?

- *Vad* ska du lära dig?
  - Grundläggande principer för programmering  
⇒ Inga förkunskaper i programmering krävs!
  - Konstruktion av algoritmer
  - Tänka i abstraktioner
  - Flera olika angreppssätt: imperativ programmering, objektorientering, funktionsprogrammering

- Programspråken Scala och Java
- Utvecklingsverktyg (editor, kompilator, utvecklingsmiljö)
- Implementera, testa, felsöka
- *Hur* ska du lära dig?
  - Genom praktiskt eget arbete: **Lära genom att göra!**
  - Genom studier av kursens teori: **Skapa förståelse!**
  - Genom samarbete med dina kurskamrater: **Gå djupare!**

### 0.0.1 hej

Denna text hamnar bara i kompediet

Hejsan svejsan

- some item

hej kod

- item
- item
- item
- item

# **Anvisningar**

**Föreläsningar**

**Övningar**

**Laborationer**

**Resurstider**

**Kontrollskrivning**

**Tentamen**





# Hur lära att programmera?

Lära genom att göra

Vilken är din lärandestil?



## **Hur bidra till kursmaterialet?**



# **Del II**

## **Moduler**



# Kapitel 1

## Introduktion

- om kursen
- sekvens
- alternativ
- repetition
- abstraktion
- programmeringsparadigmer
- editera-kompilera-exekvera
- datorns delar
- virtuell maskin
- värde
- uttryck
- variabel
- typ
- tilldelning
- val
- var
- alternativ
- if
- else
- true
- false
- logik
- MinValue
- MaxValue
- aritmetik logik
- de Morgans lagar

### 1.1 Vad är programmering?

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et

netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

## 1.2 Hur fungerar en dator?

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

## 1.3 Uttryck, värde och typ

### 1.3.1 Uttryck med enkla typer

- stränguttryck
- teckenuttryck
- aritmetiska uttryck
  - heltalsuttryck, heltalsdivision
  - heltalsrest
  - flyttalsuttryck
- logiska uttryck

### 1.3.2 Alternativuttryck

## 1.4 Variabler, val, var



## 1.5 Övning: expressions

### Mål

- Lär dig detta
- Lär dig och detta

### Förberedelser

- Läs kap. ??
- Säkerställ att du kan använda de grundläggande terminalkommandona `ls`, `cd`, `rm` och `mkdir` för att inspektera, navigera i, och manipulera filträdet, se kap. ??.
- Du behöver en dator med scala installerad. Om du inte har Scala installerad på din maskin, se installationsanvisningar i kap. ??
- Starta den editor du vill använda under övningarna, se kap. ??.

### 1.5.1 Grundläggande uppgifter

**Uppgift 1.** Starta Scala REPL och skriv satsen `println("hej")` och tryck på *Enter*.

```
$ scala
Welcome to Scala version 2.11.7 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_66).
Type in expressions to have them evaluated.
Type :help for more information.

scala> println("hejsan REPL")
```

a) Vad händer?

b) Skriv samma sats igen men "glöm bort" att skriva högerparentesen innan du trycker på *Enter*. Vad händer?

c) Evaluera uttrycket `"gurka" + "tomat"` i REPL. Vad har uttrycket för värde och typ? Vilken siffra står efter ordet `res` i variabeln som lagrar resultatet?

```
scala> "gurka" + "tomat" //tryck ENTER
```

d) Evaluera uttrycket `res0 * 42` men byt ut 0:an mot siffran efter `res` i utskriften vid förra evalueringen. Vad har uttrycket för värde och typ?

```
scala> res2 * 42
```

**Uppgift 2.** Skapa med hjälp av en editor en fil med namn `hello-script.scala` som innehåller denna enda rad:

```
println("hej skript")
```

Spara filen och kör kommandot `scala hello-script.scala` i terminalen:


```
$ scala hello-script.scala
```

- a) Vad händer?
- b) Ändra i filen så att högerparentesen saknas. Spara och kör skriptfilen igen. Vad händer?

**Uppgift 3.** Skapa med hjälp av en editor en fil med namn `hello-app.scala` som innehåller dessa rader:

```
object Hello {  
  def main(args: Array[String]): Unit = {  
    println("hejsan appen")  
  }  
}
```

Kompilera och kör koden.

- a) Vilket alternativ går snabbast att köra igång, ett skript eller en kompilerad applikation? Varför? Vilket alternativ kör snabbast när väl exekveringen är igång? 
- b) Ändra i din kod så att kompilatorn ger följande felmeddelande:  
Missing closing brace

**Uppgift 4.** Vad är en *literal*? Vilken typ har följande literaler?

- a) 42
- b) 42L
- c) '\*'
- d) "\*"
- e) 42.0
- f) 42.0F
- g) true
- h) false

**Uppgift 5.** Vad är det för skillnad på en sats och ett uttryck? 

**Uppgift 6.** Vilken typ och vilket värde har följande uttryck?

- a) `1 + 41`
- b) `1.0 + 41`
- c) `"gurk" + 'a'`
- d) `'A'`
- e) `'A'.toInt`
- f) `'0'.toInt`
- g) `'1'.toInt`
- h) `'9'.toInt`
- i) `('A' + '0').toChar`

j) `"*!%#".charAt(0)`

### Uppgift 7. Heltalsdivision

**Uppgift 8. Heltalsomfång.** För var och en av heltalstyperna i deluppgifterna nedan: undersök i REPL med operationen `MaxValue` resp. `MinValue`, till exempel `Int.MaxValue` vad som är största och minsta värde.

- a) Byte
- b) Short
- c) Int
- d) Long

**Uppgift 9.** Klassen `java.lang.Math` och paketobjektet `scala.math`.

a) Undersök genom att trycka på Tab-tangenten efter att du skriver nedan, vilka funktioner som finns i `Math` och `math`. Vad heter konstanten  $\pi$  i `java.lang.Math` respektive `scala.math`?

```
scala> java.lang.Math. //tryck TAB
scala> scala.math.    //tryck TAB
```

b) Undersök dokumentationen för klassen `java.lang.Math` här:

<https://docs.oracle.com/javase/8/docs/api/java/lang/Math.html>

Vilka trigonometriska funktioner erbjuds? Vad gör `Math.hypot`?

c) Undersök källkoden för paketobjektet `scala.math` här:

<https://github.com/scala/scala/blob/v2.11.7/src/library/scala/math/package.scala> Ge exempel på en funktion i `java.lang.Math` som inte finns i `scala.math`.

**Uppgift 10.** Vad händer här? Notera undantag (eng. *exceptions*) och nogrannhetsproblem.

- a) `Int.MaxValue + 1`
- b) `1 / 0`
- c) `1E8 + 1E-8`
- d) `1E9 + 1E-9`
- e) `math.pow(math.hypot(3,6), 2)`
- f) `1.0 / 0`
- g) `(1.0 / 0).toInt`
- h) `math.sqrt(-1)`

## 1.5.2 Extrauppgifter: öva mer på grunderna

## 1.5.3 Fördjupningsuppgifter: avancerad nivå

**Uppgift 11.** `Integer.toBinaryString`, `Integer.toHexString`

**Uppgift 12.** 0x2a**Uppgift 13.****Uppgift 14.** BigInt, BigDecimal**Uppgift 15.** Vad händer här?

```
scala> java.lang.Math.multiplyExact(2, 42)
scala> java.lang.Math.multiplyExact(Int.MaxValue, Int.MaxValue)
```

*Tips:* Eftersom `java.lang` importeras implicit så du kan även anropa `Math.multiplyExact` direkt utan `java.lang` före.

**Uppgift 16.** Sök reda på dokumentationen för funktionen `multiplyExact` i javadoc för klassen `java.lang.Math` i JDK 8.

**Uppgift 17.** Sök i javadoc för `Math` efter förekomster av texten *"throwing an exception if the result overflows"*. Vilka fler funktioner finns i `java.lang.Math` som hjälper en att upptäcka om det blir overflow?

**Uppgift 18.** Använda Scala REPL för att undersöka konstanterna nedan. Vilket av dessa värden är negativt? Vad kan man ha för praktisk nytta av dessa värden i ett program som sysslar med flyttalsberäkningar?

- a) `java.lang.Double.MIN_VALUE`
- b) `scala.Double.MinValue`
- c) `scala.Double.MinPositiveValue`

**Uppgift 19.** För typerna `Byte`, `Short`, `Char`, `Int`, `Long`, `Float`, `Double`: Undersök hur många bitar som behövs för att representera varje typs omfång?

*Tips:* Några användbara uttryck:

```
Integer.toString(Integer.MAX_VALUE + 1).size
```

```
Integer.toString((math.pow(2,16) - 1).toInt).size
```

`1 + math.log(Long.MaxValue)/math.log(2)` Se även språkspecifikationen för Scala, kapitlet om heltalslitteraler:

<http://www.scala-lang.org/files/archive/spec/2.11/01-lexical-syntax.html#integer-literals>

## 1.6 Laboration: Quiz

### Mål

- Lär dig detta
- Lär dig och detta

### Förberedelser

- Läs detta
- Läs och detta

### 1.6.1 Obligatoriska uppgifter

**Uppgift 1.** Gör först detta

**Uppgift 2.** Gör sedan detta

### 1.6.2 Frivilliga extrauppgifter

**Uppgift 3.** Gör först detta

**Uppgift 4.** Gör sedan detta



# Kapitel 2

## Kodstrukturer

- while-sats
- for-sats
- algoritmer: min/max
- summering
- swap
- paket
- import
- filstruktur
- jar
- dokumentation
- programlayout
- JDK
- konstanter vs föränderlighet
- objektorientering
- klasser
- objekt
- referensvariabler
- referenstilldelning
- anropa metoder
- block
- namnsynlighet SimpleWindow

### 2.1 Vad är programmering?

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabi-

tur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

## 2.2 Hur fungerar en dator?

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.



## 2.3 Övning: Hello

### Mål

- Lär dig detta
- Lär dig och detta

### Förberedelser

- Läs detta
- Läs och detta

### 2.3.1 Grundläggande uppgifter

#### Avdelning ditten

**Uppgift 1.** Starta Scala REPL och skriv ut en sträng. Om du inte har Scala installerad på din maskin, se installationsanvisningar i Kapitel ??

```
$ scala
Welcome to Scala version 2.11.7 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_66).
Type in expressions to have them evaluated.
Type :help for more information.

scala> println("hej")
```

a) Vad händer?


b) Vad händer sen?

**Uppgift 2.** Gör sedan detta och detta.

a) Vad händer?

b) Vad händer sen?

**Uppgift 3.** Gör sedan detta och detta.

 **Uppgift 4.** Gör sedan detta och detta med papper och penna.

#### Avdelning datten

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

### 2.3.2 Extrauppgifter: öva mer på grunderna

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

### 2.3.3 Fördjupningsuppgifter: avancerad nivå

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

## 2.4 Laboration: Quiz

### Mål

- Lär dig detta
- Lär dig och detta

### Förberedelser

- Läs detta
- Läs och detta

### 2.4.1 Obligatoriska uppgifter

**Uppgift 1.** Gör först detta

**Uppgift 2.** Gör sedan detta

### 2.4.2 Frivilliga extrauppgifter

**Uppgift 3.** Gör först detta

**Uppgift 4.** Gör sedan detta



# Kapitel 3

## Funktioner, Objekt

- parameter
- returtyp
- värdeandrop
- namnanrop
- namngivna parametrar
- aktiveringspost
- rekursion
- basfall
- anropsstacken
- objektheapen
- objekt
- modul
- def
- lazy val
- aritmetik
- slumptal

### 3.1 Vad är programmering?

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

## 3.2 Hur fungerar en dator?

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

## 3.3 Övning: Hello

### Mål

- Lär dig detta
- Lär dig och detta

### Förberedelser

- Läs detta
- Läs och detta

### 3.3.1 Grundläggande uppgifter

#### Avdelning ditten

**Uppgift 1.** Starta Scala REPL och skriv ut en sträng. Om du inte har Scala installerad på din maskin, se installationsanvisningar i Kapitel ??

```
$ scala
Welcome to Scala version 2.11.7 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_66).
Type in expressions to have them evaluated.
Type :help for more information.


scala> println("hej")
```

- a) Vad händer?
- b) Vad händer sen?

**Uppgift 2.** Gör sedan detta och detta.

- a) Vad händer?
- b) Vad händer sen?

**Uppgift 3.** Gör sedan detta och detta.

 **Uppgift 4.** Gör sedan detta och detta med papper och penna.

#### Avdelning datten

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

### 3.3.2 Extrauppgifter: öva mer på grunderna

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

### 3.3.3 Fördjupningsuppgifter: avancerad nivå

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.



## 3.4 Laboration: Quiz

### Mål

- Lär dig detta
- Lär dig och detta

### Förberedelser

- Läs detta
- Läs och detta

### 3.4.1 Obligatoriska uppgifter

**Uppgift 1.** Gör först detta

**Uppgift 2.** Gör sedan detta

### 3.4.2 Frivilliga extrauppgifter

**Uppgift 3.** Gör först detta

**Uppgift 4.** Gör sedan detta



# Kapitel 4

## Datastrukturer

- tupler
- case-klasser
- case-object i Scala vs enum i java Array
- Map
- List
- Vector
- föränderlighet
- iterering
- vektorer i Java vs Scala
- Complex
- Rational filer
- Source.fromFile
- java.nio.file

### 4.1 Vad är programmering?

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

## 4.2 Hur fungerar en dator?

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

## 4.3 Övning: Hello

### Mål

- Lär dig detta
- Lär dig och detta

### Förberedelser

- Läs detta
- Läs och detta

### 4.3.1 Grundläggande uppgifter

#### Avdelning ditten

**Uppgift 1.** Starta Scala REPL och skriv ut en sträng. Om du inte har Scala installerad på din maskin, se installationsanvisningar i Kapitel ??

```
$ scala
Welcome to Scala version 2.11.7 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_66).
Type in expressions to have them evaluated.
Type :help for more information.

scala> println("hej")
```

a) Vad händer?


b) Vad händer sen?

**Uppgift 2.** Gör sedan detta och detta.

a) Vad händer?

b) Vad händer sen?

**Uppgift 3.** Gör sedan detta och detta.

 **Uppgift 4.** Gör sedan detta och detta med papper och penna.

#### Avdelning datten

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

### 4.3.2 Extrauppgifter: öva mer på grunderna

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

### 4.3.3 Fördjupningsuppgifter: avancerad nivå

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

## 4.4 Laboration: Quiz

### Mål

- Lär dig detta
- Lär dig och detta

### Förberedelser

- Läs detta
- Läs och detta

### 4.4.1 Obligatoriska uppgifter

**Uppgift 1.** Gör först detta

**Uppgift 2.** Gör sedan detta

### 4.4.2 Frivilliga extrauppgifter

**Uppgift 3.** Gör först detta

**Uppgift 4.** Gör sedan detta





# **Del III**

## **Appendix**



# Appendix A

## Terminalfönster och kommandoskal

### A.1 Vad är ett terminalfönster?

I ett terminalfönster kan man skriva kommandon som till exempel kör program och hanterar filer på din dator. När man programmerar använder man ofta terminalkommando för att kompilera och exekvera sina program.

#### Terminal i Linux

#### PowerShell i Microsoft Windows

Microsoft Windows är inte Unix-baserat, men i kommandotolken PowerShell finns alias definierat för en del vanliga unix-kommandon. Du startar Powershell t.ex. genom att trycka på Windows-knappen och skriva powershell.

#### Terminal i Apple OS X

Apple OS X är ett Unix-baserat operativsystem. Många kommandon som fungerar under Linux fungerar också under Apple OS X.

### A.2 Några viktiga terminalkommando



# **Appendix B**

## **Editera**

**B.1 Vad är en editor?**

**B.2 Välj editor**



# Appendix C

## Kompilera och exekvera

### C.1 Vad är en kompilator?

### C.2 Java JDK

#### C.2.1 Installera Java JDK

### C.3 Scala

#### C.3.1 Installera Scala-kompilatorn

### C.4 Read-Evaluate-Print-Loop (REPL)

För många språk, t.ex. Scala och Python, finns det en interaktiv tolk som gör det möjligt att exekvera enstaka programrader och direkt se effekte. En sådan tolk kallas Read-Evaluate-Print-Loop eftersom den läser en rad i taget och översätter till maskinkod som körs direkt.

#### C.4.1 Scala REPL

##### Kommandon i REPL

`:paste`

Kortkommandon: Ctrl+K etc.





# **Appendix D**

## **Dokumentation**

**D.1 Vad gör ett dokumentationsverktyg?**

**D.2 scaladoc**

**D.3 javadoc**



# **Appendix E**

## **Integrerad utvecklingsmiljö**

### **E.1 Vad är en IDE?**

### **E.2 ScalalDE och Eclipse**

#### **E.2.1 Installera ScalalDE**

### **E.3 Handledning ScalalDE/Eclipse**



# **Appendix F**

## **Byggverktyg**

**F.1 Vad gör ett byggverktyg?**

**F.2 Byggverktyget sbt**

Installera sbt



# **Appendix G**

## **Versionshantering**

### **G.1 Vad är versionshantering?**

### **G.2 Versionshanteringsverktyget git**

**Installera git**





## **Appendix H**

### **Lösningsförslag till övningar**

## H.1 Övning: hello

### H.1.1 Grundläggande uppgifter

#### Uppgift 1.

a) Följande skrivs i terminalen: hello repl

```
$ scala
Welcome to Scala version 2.11.7 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_66).
Type in expressions to have them evaluated.
Type :help for more information.

scala> println("hej repl")
hej repl

scala>
```

b) Scala REPL väntar på att raden ska avslutas, vilket indikeras med vertikalstreck | på en ny rad:

```
scala> println("hej repl")
|
```

#### Uppgift 2.

a) Vad händer?

b) Vad händer sen?

**Uppgift 3.** Gör sedan detta och detta.

**Uppgift 4.** Gör sedan detta och detta med papper och penna.



### H.1.2 Extrauppgifter: öva mer på grunderna

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

### H.1.3 Fördjupningsuppgifter: avancerad nivå

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

## H.2 Övning: hello

### H.2.1 Grundläggande uppgifter

**Uppgift 1.** a) Detta hände. 42. b) Detta hände. 43.

**Uppgift 2.** a) Vad händer? b) Vad händer sen?



**Uppgift 3.** Gör sedan detta och detta. **Uppgift 4.** Gör sedan detta och detta med papper och penna.

### H.2.2 Extrauppgifter: öva mer på grunderna

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

### H.2.3 Fördjupningsuppgifter: avancerad nivå


Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

## H.3 Övning: hello

### H.3.1 Grundläggande uppgifter

**Uppgift 1.** a) Detta hände. 42. b) Detta hände. 43.

**Uppgift 2.** a) Vad händer? b) Vad händer sen?

**Uppgift 3.** Gör sedan detta och detta. **Uppgift 4.** Gör sedan detta och detta   
med papper och penna.

### H.3.2 Extrauppgifter: öva mer på grunderna

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

### H.3.3 Fördjupningsuppgifter: avancerad nivå

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

# **Appendix I**

## **Ordlista**