

Programmering, grundkurs pgk

Föreläsningsanteckningar pgk (EDAA45)

Björn Regnell

Datavetenskap, LTH

Lp1-2, HT 2016

1 Introduktion

2 Kodstruktur

1 Introduktion

- Om kursen
- Meddelande från Code@LTH
- Om kursen
- Meddelande från Code@LTH

2 Kodstruktur

Om kursen

Veckoöversikt

<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	–	–

Vad lär du dig?

- Grundläggande principer för programmering:
Sekvens, Alternativ, Repetition, Abstraktion (SARA)
⇒ Inga förkunskaper i programmering krävs!
- Konstruktion av algoritmer
- Tänka i abstraktioner
- Förståelse för flera olika angreppssätt:
 - **imperativ programmering**: satser, föränderlighet
 - **objektorientering**: inkapsling, återanvändning
 - **funktionsprogrammering**: uttryck, oföränderlighet
- Programspråken **Scala** och **Java**
- Utvecklingsverktyg (editor, kompilator, utvecklingsmiljö)
- Implementera, testa, felsöka

Hur lär du dig?

- Genom praktiskt **eget arbete**: **Lära genom att göra!**
- Genom studier av kursens teori: **Skapa förståelse!**
- Genom samarbete med dina kurskamrater: **Gå djupare!**

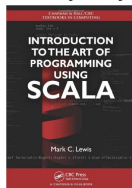
Kurslitteratur



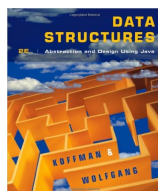
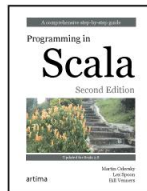
- **Kompedium** med föreläsningbilder, övningar & laborationer
- Säljs på KFS
<http://www.kfsab.se/>

Rekommenderade böcker

För nybörjare:



För de som redan kodat en del:



Meddelande från **Code@LTH**

Om kursen

Veckoöversikt

<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	–	–

Vad lär du dig?

- Grundläggande principer för programmering:
Sekvens, Alternativ, Repetition, Abstraktion (SARA)
⇒ Inga förkunskaper i programmering krävs!
- Konstruktion av algoritmer
- Tänka i abstraktioner
- Förståelse för flera olika angreppssätt:
 - **imperativ programmering**: satser, föränderlighet
 - **objektorientering**: inkapsling, återanvändning
 - **funktionsprogrammering**: uttryck, oföränderlighet
- Programspråken **Scala** och **Java**
- Utvecklingsverktyg (editor, kompilator, utvecklingsmiljö)
- Implementera, testa, felsöka

Hur lär du dig?

- Genom praktiskt **eget arbete**: **Lära genom att göra!**
- Genom studier av kursens teori: **Skapa förståelse!**
- Genom samarbete med dina kurskamrater: **Gå djupare!**

TESTSLAJD EJ I KOMPENDIUM

- *Hej* på dig
- blablab
- blabla

```
hej kod
```

Meddelande från Code@LTH

1 Introduktion

2 Kodstruktur

- Att göra denna vecka
- Algoritmer

Resurstider och Labbar

- Laborationer är **obligatoriska**.
Ev. sjukdom måste anmälas **före** till kursansvarig!
- Resurstiderna hade närvaro på endast ca. 50%.
Varför?

Att göra i Vecka 1: Fatta kodstruktur

- 1 Läs följande kapitel i kursboken:
2.1-2.6, 4, 5.4, 7.2, 7.5-7.6, 7.8-7.9
Begrepp: algoritm, pseudokod, abstraktion, oändlig loop, while-sats, for-sats, paket, import, referensvariabel, objekt, referenstilldelning, referenslikhet
- 2 Gör övning 2: Paket, kodfiler, och dokumentation
- 3 OBS! Ingen lab denna vecka
- 4 Träffas i samarbetsgrupper och hjälp varandra att förstå
- 5 Gör klart **samarbetskontrakt** och visa för handledare på resurstid
- 6 **Koda på resurstiderna** och få hjälp och tips!
Varför var de så få kom kom till resurstiderna vecka 1?

Vad är en algoritm?

En **algoritm** är en sekvens av instruktioner som beskriver hur man löser ett problem

Exempel:
matrecept

Vad är en algoritm?

En **algoritm** är en sekvens av instruktioner som beskriver hur man löser ett problem

Exempel:

matrecept

uppdatera highscore i ett spel

...

XX