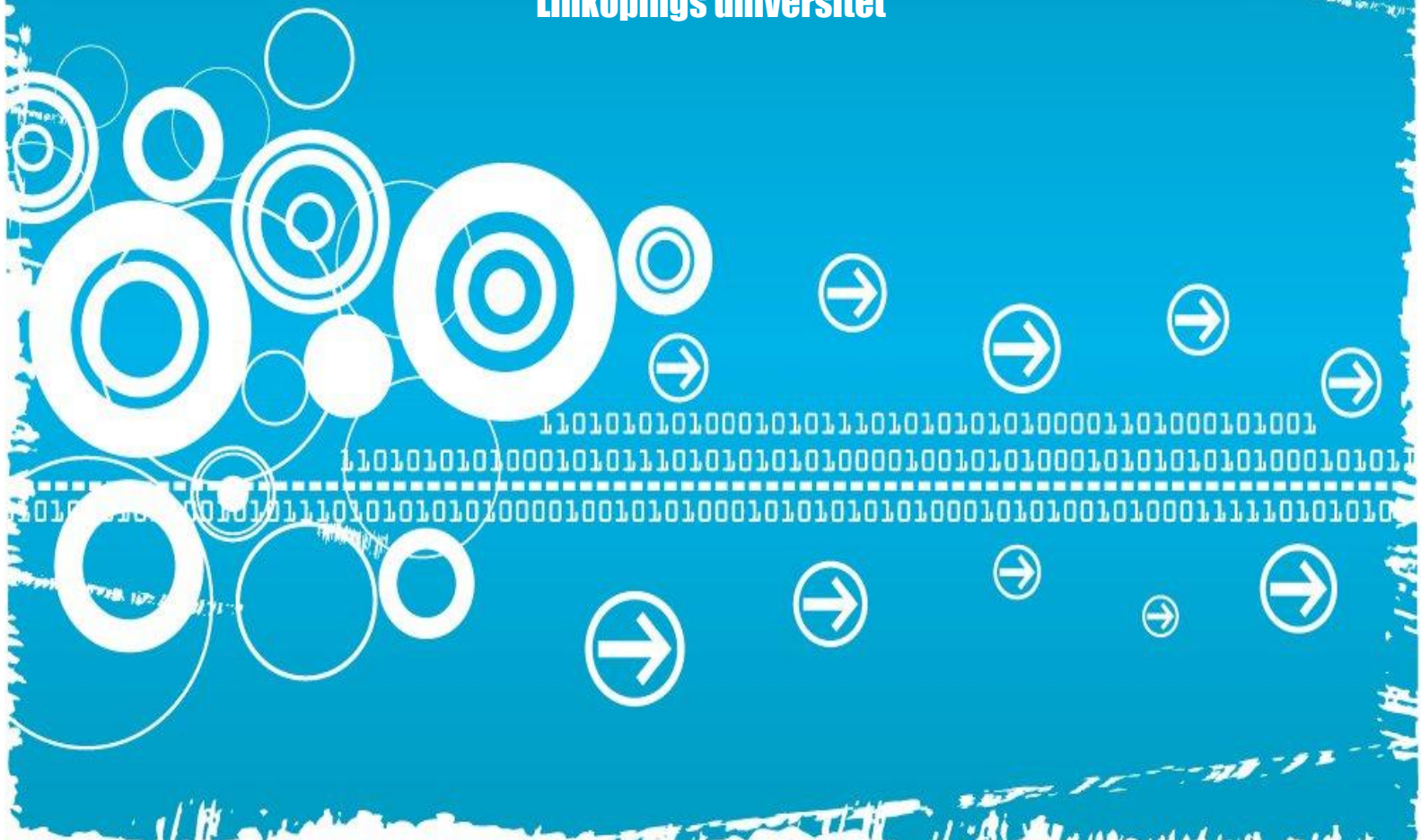


TDDC77 - Objektorienterad programmering

Laboration 2

**Institutionen för datavetenskap
Linköpings universitet**



2.1 Kom igång med Eclipse

Från och med nu ska du utveckla Java-program i ett program(stor editor) som heter *Eclipse*. *Eclipse* är en integrerad utvecklingsmiljö, IDE, som erbjuder många fördelar när ni programmerar, bland annat syntax highlighting, inkrementell kompilering (kompilering görs hela tiden i bakgrunden), samt stöd för debugging och refactoring. *Eclipse* är dessutom världens mest populära miljö för Java så chansen är stor att ni kommer arbeta med *Eclipse* i framtiden. *Eclipse* är gratis och finns för Windows, Mac, Linux med mera. Alltså går det utmärkt att arbeta hemma, gå bara till www.eclipse.org och ladda ner rätt version för din dator. Från terminalen kan ni starta *Eclipse* med:

```
> /opt/eclipse/4.10/eclipse&
```

Ni kommer att bli tillfrågade om var ni vill spara er Eclipse-arbetsyta (workspace). Välj `er_hemkatalog/workspace` och tryck OK.

Eclipse tutorial:

<https://www.vogella.com/tutorials/Eclipse/article.html>

Eclipse Documentation:

<https://help.eclipse.org/latest/index.jsp?topic=%2Forg.eclipse.jdt.doc.user%2FgettingStarted%2Fqs-3.htm>

2.2 GissaTal - ett litet spel*

Nu över till en lite mer intressant uppgift. Ni ska programmera ett litet spel som heter GissaTal. Första spelaren matar in ett heltal mellan 0 och 100. Sen får den andra spelaren gissa vilket tal det var. För varje gissning får andra spelare reda på om gissningen var rätt, för liten eller för stor.

1. Skapa ett program som ber spelaren att mata in ett heltal mellan 0 och 100. Därefter ska programmet tömma skalfönstret genom att skriva ut radbrytningar tillräckligt många gånger.
2. Nu ska programmet be spelare två att gissa på ett heltal mellan 0 och 100. Om gissningen är fel så ska programmet be om ett nytt heltal. Om gissningen är rätt ska programmet gratulera till vinsten, skriva ut antalet gissningar som behövdes för att hitta rätta talet och sen avslutas.
3. Sist ska ni införa ledtrådar. Spelare två ska få veta om hans/hennes gissningar för liten eller för stor.

En programkörning efter steg 3 skulle kunna se ut så här:

```
[john@emil23 java]$ java GissaTal
Spelare ett, Mata in ett heltal mellan 0 och 100: 18
*** Skärmen töms ***
Spelare två, chansa på ett heltal mellan 0 och 100: 20
För stort!
Spelare två, chansa på ett heltal mellan 0 och 100: 5
För litet!
Spelare två, chansa på ett heltal mellan 0 och 100: 15
För litet!
Spelare två, chansa på ett heltal mellan 0 och 100: 18
Rätt! Du behövde 4 gissningar.
[john@emil23 java]$
```

Kör programmet för handledaren. Kan ni komma på någon effektiv gissningsstrategi? Hur många gissningar behöver man maximalt med er strategi? Skulle man kunna skriva ett Javaprogram som spelade med er strategi, dvs där en människa matar in ett heltal och sen låter programmet försöka hitta talet med så få gissningar som möjligt (utan att fuska)? Prova!

2.3 Textstatistik - ett ordbehandlingsverktyg*

Förr i tiden skrev man långa dokument på skrivmaskin (har ni provat någon gång?). Fördelarna med att skriva på dator är många och stora. Några av fördelarna är att datorn mycket snabbt kan gå igenom texten och hjälpa till med stavning, grammatik, avstavning och statistik.

Det sistnämnda ska ni nu prova på att programmera - ett program som tar fram statistik på antal ord, antal bokstäver, antal skiljetecken samt medellängd på orden. En körning av programmet skulle kunna se så här ut:

```
[john@emil23 java]$ java TextStatics this is an example
Antal ord: 4
Antal bokstäver: 15
Antal skiljetecken: 3
Medellängd på ord: 3.75 bokstäver
[john@emil23 java]$
```