

## Praktisk information

N/A.

## Emner til lektion 3 (20/9)

Loops og Conditions.

Definition af egne klasser.

Obligatorisk Opgave.

## Forberedelse til lektionen

Diverse *If-else*, *switch*, *while* og *for* -strukturer burde i hovedtræk være kendt fra fx JavaScript.

Kig PowerPoint præsentationen til [kap 4](#) igennem og notér ned, hvis der er emner der skal uddybes.

**NB:** Slide 34 – 43, *Comparing Data* er det nødvendigt at i kender.

Slide 68 – 72, *Iterators* har i sandsynligvis ikke før stødt på (bortset fra at `java.util.Scanner` er en *Iterator*). Det er ikke vigtigt lige nu, hvorfor slide 88 – 89 heller ikke er vigtige.

Studer kap 5, afsnit 5.1 – 5.5 meget grundigt. Indlæs eksemplerne fra teksten i NetBeans og eksperimenter med dem.

Løs *PP5.1* og *PP5.2* side 238 (den *Coin*-klasse der skal benyttes er *Listning 5.4* side 186), samt *PP5.4*.

Afsnit 5.6 skal blot skimmes. Dog skal underafsnittet om *this*-referancen side 211-212 kendes.

Afsnit 5.7 og 5.8 skal blot skimmes.

Kom så langt som muligt med 1. del af den obligatoriske opgave.

## Bearbejdelse af dagens emner

Vi tager en kort snak om emnerne i kapitel 4. Eventuelle uklarheder forsøges afhjulpet. For en sikkerheds skyld tager vi en enkelt opgave i klassen (HiLo).

Det grundlæggende stof om klasser og objekter (5.1 – 5.5, sl 1-68) diskuteres og vi kigger på nogle af bogens Programming Projects og eksempler undervejs:

- *SnakeEyes* (brug din egen terning fra sidste lektion)
- *Coinflip*: Eksempler og løsninger til *PP5.1* og *PP5.2*
- *Løsninger til PP5.4*
- *I klassen: PP5.5*
- *Teori (til s40)*
- *Static members*

Resten af kapitlet gennemgås overfladisk.

Arbejde på Matadorspillet.

## Emner til lektion 4 (27/9)

Arv og Polymorphism.

## Forberedelse til næste lektion

Løs første del af den obligatoriske opgave.

Fra kapitel 8:

- Eksperimenter grundigt med eksemplerne i afsnit 8.1 og 8.2 (Listing 8.1 – 8.9). Det er vigtigt at blive fortrolige med begreberne *superclass/subclass*, *extends (arver fra)*, *protected*, *super()* (*kald af constructor i superclass*) og *super.someMethod()* (*kald af metode i superclass, som ellers er overskrevet*).
- Studer afsnit 8.3 og 8.4 grundigt.
- Læs afsnit 8.5 og tjek jeres forståelse med listen på side 402.
- Løs PP8.1 side 406 og PP8.4 side 407.

Matador:

- [Opgavebeskrivelsen på BlackBoard](#) er nu opdateret med informationer om arbejdet til lektion 4 og 5.