# Kotlin

<https://datsoftlyngby.github.io/dat4sem2020spring/AP>

(Der skulle ligge en Powerpoint)

Procedural programmering – processer kalder processer

OOP

Funktionel programmering (functions as first-class members)

Pragmatism = Readability + robustness

Strong but discrete types

Keywords: val, var, fun, when/is, companion object, data class, operator overrides, infix, open)

Coroutines = asynkrone, non-blocking processer? (kw: suspend)

# Trådprogrammering

**Trådproblematik:**

(i = i + 1) => højre udregning først, dernæst assignes værdien til variablen. Hvis to tråde tilgår dette samtidig kan de derfor returnere samme værdi. Eks. hvis i==7 => Tråd1: (i=i+1) Tråd2: (i=i+1) Slutresultat: (i==8).

## Løsningsværktøjer til diverse trådproblemer:

**Synchronized blocks:**

Kan kun tilgås af én tråd ad gangen.

Synchronized(this) betyder, at *this* er et **monitor object** (”bamsen”). Only one thread can execute inside a Java code block synchronized on the same monitor object.  
Synchronized metoder lukker bruger internt *synchronized(this).*

**wait() + notify() / notifyAll() :**

* eksempel:

synchronized void put(E item) throws InterruptedException {

while (bufferIsFull) {

wait();

}

buffer.put(item);

notifyAll();

}

* Venter en tråd, slipper den sit dips på en synchronized blok og lader andre komme til i mellemtiden. På den måde kan mange tråde ende i en kø til at tilgå alt efter wait(). NotifyAll giver trådene lov til at fortsætte.
* Wait bør for en sikkerheds skyld altid være i en while blok, fordi JavaVirtualMachine kan i princippet afbryde tråden når som helst (det står i specifikationen).

**Deadlocks**:  
Hvis synkroniseringer i flere blokke gør at tråde ikke kan komme videre fordi de venter på hinanden.

**PULL & PUSH-blokke:** Hvad er det, hvad skal jeg bruge det til, hvad lærer man af det?

# Alt Godt Fra Havet

**Function<From, To>:** Callbacks i Java. Egentlig en javaklasse, men kan bruges som en generic metode, der kan gives og returneres af andre metoder.

**Predicate<E>:** Functions som returnerer en boolean.